

**Power
Week**

Université IBM i 2019



22 et 23 mai

IBM Client Center Paris

S52 – CCSID et Unicode

Christian Massé
Volubis
cmasse@volubis.fr



**Power
Week**

Université IBM i 2019



22 et 23 mai

IBM Client Center Paris



Volubis.fr

**Conseil et formation sur OS/400, I5/OS puis IBM i
depuis 1994 !**

Dans nos locaux, vos locaux ou par Internet

The background of the slide features a complex network diagram. It consists of numerous small, light gray circular nodes connected by thin, light gray lines. The nodes are distributed across the entire page, creating a dense web of connections. The overall appearance is that of a digital or social network.

CCSID

CCSID



- A l'origine, sur nos machines, était le CCSID.
- Avec un jeu de caractères donné ([Latin-1](#), [cyrillique](#), [hébreu](#) par ex.) définition d'une grille de codification de tous les caractères pour un pays donné :
- Nous avons le même jeu de caractères que les espagnols, nous n'avons pas le même code page (codification).
- **Le CCSID est la codification du jeu de caractères et du code page d'origine, d'une donnée.**
- "Cette données est Française !" (CCSID au niveau zone, venant du CCSID du fichier, lui même venant du job de création)
- si le CCSID du job (qui lit le fichier) est différent (il vient de la langue de l'utilisateur, sinon de QCCSID), il faut modifier la valeur héxa afin que le "é" Français s'affiche é pour l'utilisateur Espagnol ou Danois et non "{" (par exemple).
- La codification [UNICODE](#) est une codification sur plusieurs octets permettant dans une même grille, de coder **tous les caractères du monde**
(y compris chinois, bengali, braille, symboles mathématiques et notes de musique !)



- 65535 ?

C'est simple, ça veut dire : « ne t'occupe pas de ça ! »

C'est toujours une mauvaise idée

C'est compliqué d'en sortir (à quel point avez-vous *bidouillé* pour compenser le fait que l'OS ne faisait pas le job ?)

The background of the slide is a complex network diagram. It consists of numerous small, light-grey circular nodes scattered across the frame. These nodes are interconnected by a dense web of thin, light-grey lines, creating a mesh-like structure that resembles a neural network or a data connectivity map. The overall aesthetic is clean and technical.

UNICODE

Définition



- Le standard Unicode est constitué d'un répertoire de plus de 110 000 caractères couvrant 100 écritures, d'un ensemble de tableaux de codes pour référence visuelle (braille), d'une méthode de codage et de plusieurs codages de caractères standard, d'une énumération des propriétés de caractère (lettres majuscules, minuscules, symboles, ponctuation, etc.) d'un ensemble de fichiers de référence des données informatiques, et d'un certain nombre d'éléments liés, tels que des règles de normalisation, de décomposition, de tri, de rendu et d'ordre d'affichage bidirectionnel (pour l'affichage correct de texte contenant à la fois des caractères d'écritures droite à gauche, comme l'arabe et l'hébreu, et de gauche à droite).
- En pratique, Unicode reprend intégralement la norme [ISO/CEI 10646](#), puisque cette dernière ne normalise que les caractères individuels en leur assignant **un nom** et un numéro normatif (appelé *point de code*) et une description informative très limitée.

Définition



Voici une toute petite partie des tables UNICODE (les nombres sont présentés en notation hexadécimale):

	000	001	002	003	004	005	006	007
0	0	@	P	`	p
1	!	1	A	Q	a	q
2	"	2	B	R	b	r
3	#	3	C	S	c	s
4	\$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	'	7	G	W	g	w
8	(8	H	X	h	x
9)	9	I	Y	i	y
A	*	A	J	Z	j	z
B	+	B	[k		
C	,	C	<	L	\	
D	-	D	=	M]]
E	E	>	N	^	~
F	/	F	?	O	_	o

Caractères Unicode
0000 à 007F (0 à 127)
(caractères latins)

	008	009	00A	00B	00C	00D	00E	00F
0	À	Á	Â	Ã	Ä	Å
1	Æ	Ç	Ð	Ñ	Ò	Ó
2	Ô	Õ	Ö	×	Ø	Ù
3	Ú	Û	Ü	Ý	ÿ	...
4	à	á	â	ã	ä	å
5	æ	ç	ð	ñ	ò	ó
6	ô	õ	ö	ø	ù	ú
7	û	ü	ý	ÿ
8	À	Á	Â	Ã	Ä	Å
9	Æ	Ç	Ð	Ñ	Ò	Ó
A	Ô	Õ	Ö	×	Ø	Ù
B	Ú	Û	Ü	Ý	ÿ	...
C	à	á	â	ã	ä	å
D	æ	ç	ð	ñ	ò	ó
E	ô	õ	ö	ø	ù	ú
F	û	ü	ý	ÿ

Caractères Unicode
0080 à 00FF (128 à 255)
(caractères latins, dont accentués)

	090	091	092	093	094	095	096	097
0	ए	ठ	र	ी	ऋ	ॠ	ॡ	ॢ
1	ॣ	।	॥	०	१	२	३	४
2	५	६	७	८	९	०	१	२
3	३	४	५	६	७	८	९	०
4	३	४	५	६	७	८	९	०
5	३	४	५	६	७	८	९	०
6	३	४	५	६	७	८	९	०
7	३	४	५	६	७	८	९	०
8	३	४	५	६	७	८	९	०
9	३	४	५	६	७	८	९	०
A	३	४	५	६	७	८	९	०
B	३	४	५	६	७	८	९	०
C	३	४	५	६	७	८	९	०
D	३	४	५	६	७	८	९	०
E	३	४	५	६	७	८	९	०
F	३	४	५	६	७	८	९	०

Caractères Unicode
0900 à 097F (2304 à 2431)
(caractères devanagari/sanskrit)

	110	111	112	113	114	115	116	117
0	ㄱ	ㅋ	ㆁ	ㆁ	ㄷ	ㅌ	ㄴ	ㄴ
1	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
2	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
3	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
4	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
5	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
6	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
7	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
8	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
9	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
A	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
B	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
C	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
D	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
E	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ
F	ㄷ	ㅌ	ㄴ	ㄴ	ㄷ	ㅌ	ㄴ	ㄴ

Caractères Unicode
1100 à 117F (4352 à 4479)
(caractères hangul jamo/coréen)

plus d'informations sur <http://www.unicode.org>.

Définition



- Dans la table des caractères Unicode on ajoute un index numérique associé à chaque caractère. Notons bien qu'il ne s'agit pas d'une représentation en mémoire, juste d'un nombre entier, appelé *point de code*. L'espace de codage de ces nombres est divisé en 17 zones de 65 536 points de codes. Ces zones sont appelées *plans*.

Le point de code est noté U+xxxx où xxxx est en hexadécimal, et comporte 4 à 6 chiffres :

- 4 chiffres pour le premier plan, appelé *plan multilingue de base* (donc entre U+0000 et U+FFFF);
 - 5 chiffres pour les 15 plans suivants (entre U+10000 et U+FFFFF);
 - 6 chiffres pour le dernier plan (entre U+100000 et U+10FFFF).
- Ainsi, le caractère nommé « **Lettre majuscule latine c cédille** » a un index de U+00C7.

Il appartient au premier plan.

En principe toutes les positions de code entre U+0000 et U+10FFFF sont disponibles, mais certains intervalles sont perpétuellement réservés à des usages particuliers, notamment une zone d'indirection exclue pour permettre le codage UTF-16 (cf. ci-dessous), les zones à usage privé et quelques régions (par exemple U+FFFE ou U+FFFF) contenant des *non-caractères* dont l'usage est interdit dans un échange de données conforme. Les autres positions de code sont soit déjà assignées à des caractères, soit réservées pour normalisation future.

UTF-8



- Techniquement, il s'agit de [coder](#) les caractères [Unicode](#) sous forme de séquences de un à quatre [codets](#) d'un [octet](#) chacun. La norme Unicode définit entre autres un ensemble (ou répertoire) de caractères. Chaque caractère est repéré dans cet ensemble par un index entier aussi appelé « point de code ». Par exemple le caractère « € » ([euro](#)) est le 8365e caractère du répertoire Unicode, son index, ou point de code, est donc 8364 (on commence à compter à partir de 0).
Le répertoire Unicode peut contenir plus d'un million de caractères, ce qui est bien trop grand pour être codé par un seul [octet](#) (limité à des valeurs entre 0 et 255). La norme Unicode définit donc des méthodes standardisées pour coder et stocker cet index sous forme de séquence d'octets : UTF-8 est l'une d'entre elles, avec [UTF-16](#), [UTF-32](#) et leurs différentes variantes.
La principale caractéristique d'UTF-8 est qu'elle est rétro-compatible avec la norme ASCII, c'est-à-dire que tout caractère ASCII se code en UTF-8 sous forme d'un unique octet, identique au code ASCII. Par exemple « A » (A majuscule) a pour code ASCII 65 et se code en UTF-8 par l'octet 65. Chaque caractère dont le point de code est supérieur à 127 (caractère non ASCII) se code sur 2 à 4 octets. Le caractère « € » (euro) se code par exemple sur 3 octets : 226, 130, et 172.

UTF-16



- L'[UTF-16](#) est un bon compromis lorsque la place mémoire n'est pas trop restreinte, car la grande majorité des caractères Unicode assignés pour les écritures des langues modernes (dont les caractères les plus fréquemment utilisés) le sont dans le plan multilingue de base et peuvent donc être représentés sur 16 bits.

Les points de code des seize plans supplémentaires nécessitent une transformation sur deux mots de 16 bits :

on soustrait 0x10000 au point de code, ce qui laisse un nombre de 20 bits dans l'étendue 0..0xFFFFF

les 10 bits de poids fort (un nombre entre 0 et 0x3FF) sont additionnés à 0xD800, et donnent la première unité de code dans la demi-zone haute (0xD800..0xDBFF)

les 10 bits de poids faible (un nombre entre 0 et 0x3FF) sont additionnés à 0xDC00, et donnent la seconde unité de code dans la demi-zone basse (0xDC00..0xDFFF)

- En résumé, il y a 2 plages réservées :

Haute : X'D800' – X'DBFF' = 4 * 256 combinaisons

Basse : X'DC00' – X'DFFF' = 4 * 256 combinaisons

soit 1024 * 1024 caractères supplémentaires

Codage UTF-16

hi \ lo	DC00	DC01	...	DFFF
D800	10000	10001	...	103FF
D801	10400	10401	...	107FF
:	/	/	...	/
DBFF	10FC00	10FC01	...	10FFFF

- codage sur 2 octets pour les 63488 premiers caractères (comme UCS-2 ci-dessous)
- codage sur 4 octets pour les autres caractères (subrogate characters)

UCS-2



UCS-2 est la version simplifiée de UTF-16 ne prévoyant pas les indirections et ne permet donc que de stocker des caractères appartenant au premier plan. (arrivée avant UTF-16, c'est un peu obsolète aujourd'hui)

- UTF-16 intègre le sens de lecture droite->gauche/gauche->droite (voir BOM), pas UCS2
- UTF-16 intègre une normalisation (un nommage) de chaque caractère, pas UCS2
- UCS2 peut être transformé en UTF-16, le contraire n'est pas forcément vrai

UTF-32

L'UTF-32 est utilisé lorsque la place mémoire n'est pas un problème (codage sur 4 octet) et que l'on a besoin d'avoir accès à des caractères de manière directe et sans changement de taille (hiéroglyphes par ex.).

BOM



- BOM (pour l'anglais *byte order mark*) ou en [Français](#) **indicateur d'ordre des octets** (IOO).

Donnée qui indique l'utilisation l' encodage unicode ainsi que l'ordre des octets, généralement situé au début de certains fichiers texte

Codage	Séquence d'octets (Représentation)
UTF-8	EF BB BF
UTF-16 Big Endian	FE FF
UTF-16 Little Endian	FF FE
UTF-32 Big Endian	00 00 FE FF
UTF-32 Little Endian	FF FE 00 00

Sur IBM i



- Sur IBM i , les différentes versions Unicode sont représentées par un code page
- **UCS-2** -> **CCSID(13488)**
- **UTF-8** -> **CCSID(1208)**
- **UTF-16 (BE)** -> **CCSID(1200)**
- **UTF-32** -> *(non implémentée)*
- Il faut parallèlement avoir installé la bibliothèque ICU (option 39 du système)

```
5770SS1 39 Composants internationaux pour Unicode
```

Sur IBM i



- Sur IBM i , les différentes versions Unicode sont représentées par un code page

Avec UCS-2 et UTF-16, une zone base de données de 20 caractères = 40 Octets
Avec UTF-8 une zone 20 caractères = 20 Octets, donc potentiellement trop courte, pour les caractères accentués
à réserver au VARCHAR, CLOB et fichiers IFS.

```
A          UCS2          10G    CCSID(13488)
A* Lg de stockage = 20 octets
A          UTF16         10G    CCSID(1200)
A* Lg de stockage = 20 octets
A          UTF8          10A    CCSID(1208)
A* Lg de stockage = 10 octets
```

impossible d'être certain de placer 10 caractères dans une zone 10A CCSID(1208)

- Avec des caractères du premier plan, c'est possible
- Mais le € prend trois octets : rappel ->

UTF-8 (hex)	0xE2 0x82 0xAC (e282ac)
UTF-8 (binary)	11100010:10000010:10101100

Bref, mieux vaut utiliser UCS-2 ou UTF-16 dans la base de données.

- **Bref, mieux vaut utiliser UCS-2 ou UTF-16 dans la base de données.**

```
create table bdvinu2/producteurs (  
  pr_code      integer      primary key      ,  
  pr_nom       nchar(50)    not null      ,  
  pr_adresse   nchar(60)    ,  
  pr_cdpst     nchar(12)    ,  
  pr_commune   nchar(50)    ,  
  pr_proprio   nchar(60)    ,  
  pr_responsable nchar(50)  ,  
  pr_tel       nchar(20)    ,  
  pr_fax       nchar(20)    ,  
  pr_vente     nchar(3)     check (pr_vente in ('oui' , 'non')) ,  
  pr_visite    nchar(3)     check (pr_visite in ('oui' , 'non')) ,  
  hr_code      integer      references hotelresto ,  
  pr_avis      nvarchar(255) ,  
  pr_cote      nchar(10)    ,  
  pr_photo     integer      ,  
  pr_carte1    nchar(10)    references cartes ,  
  pr_carte2    nchar(10)    references cartes ,  
  pr_carte3    nchar(10)    references cartes ,  
  cree        nchar(1)     check (cree in ('O' , 'N')) ,  
  mon_avis     nvarchar(50) ,  
  ma_cote      nchar(10)    ,  
  appel_code   integer      references appellations) RCFMT PRODF ;
```

nchar en SQL

Quelques questions existentielles



- **DDS**

- ET si je dois faire un test?

Testez le(s) caractère(s) de votre choix, la constante sera passée en unicode afin d'être comparée correctement à la variable

- Et si je dois faire un SST

pensez "position du caractère" (ici le 3eme) jamais en octet...

COD1

I

SST(CODE 3 1)

Commandes



- CPYF
 - fonctionne, CRTFILE(*YES) créé un fichier à l'identique (zones Unicode)

 - la commande de copie fonctionne aussi avec des CCSID différents, si FMTOPT(*MAP)
 - d'EBCDIC vers Unicode sans problème
 - de Unicode vers EBCDIC quand l'alphabet est compatible

 - le paramètre INCREL (permettant de ne copier que les champs répondant à un test) ne supporte **pas** Unicode

- OPNQRYP :
 - si vous faites des tests, vos données sont transformées en UNICODE pour pouvoir être comparées (sauf CCSID à 65535)
OPNQRYP FILE((AF4TEST/JAPONAIS)) QRYSLT('code *EQ "ABC"')

- la commande CPYFRMIPMF fonctionne, mais TOFILE(*PRINT) ne produit pas un résultat lisible

Quelques questions existentielles



- SQL
 - ET pour les vues SQL

System Name	Data Type	Length	Nullable	Generated Value	Default Value	Hidden	Text	CCSID
F1	GRAPHIC	40	No	Néant	N"			13488
CODE	GRAPHIC	3	No	Néant	N"			1200

```
CREATE VIEW AF4TEST/JAPL3 (  
    F2 )  
AS  
( SELECT TRIM(F1) CONCAT '/' CONCAT CODE AS F2 FROM AF4TEST/JAPONAIS)  
RCDFMT JAPL3 ;
```

les constantes seront, là aussi, transformées en UNICODE (expressions et/ou tests), la fonction SPACE(n) produit n caractères ' ' en Unicode, etc...

Columns Query Text

System Name	Data Type	Length	Nullable	Generated Value	Default Value	Hidden	Text	CCSID
F2	VARGRAPHIC	44	No	Néant	Aucune valeur pa...			1200

Notez que concaténer de l'UCS2 (13488) et de l'UTF-16 (1200) produit de l'UTF-16



Quelques questions existentielles

- **FTP**

- les transferts de fichiers se font suivant le type de transfert demandé :

ASCII ou TYPE A (*dft*)

fichiers texte, transformé du CCSID de DB2 vers ASCII international (819)

sauf paramétrage particulier dans CHGFTP

- EBCDIC ou TYPE B (rare)

Transfert en EBCDIC

- Bin ou TYPE I (binaire)

pas de conversion lors du transfert



Affichage et impression

- **RDI sait traiter de l'Unicode**


- pas de problèmes, dans la perspective base de données
- LPEX reconnaît les fichiers sources en Unicode
- LPEX reconnaît le type RPG **UCS2** et la fonction %UCS2()
- le debugger affiche une zone Unicode avec de l'alphabet occidental, mais pas les autres alphabets

Affichage et impression



- **RDI**

Perspective Debug



▷	*IN	
	CODECLI	00000002.
	DEPCLI	00.
	RAISOC	client DEUX
▷	*IN	
	CODECLI	00000006.
	DEPCLI	00.
	RAISOC
	VILLE	PEKIN

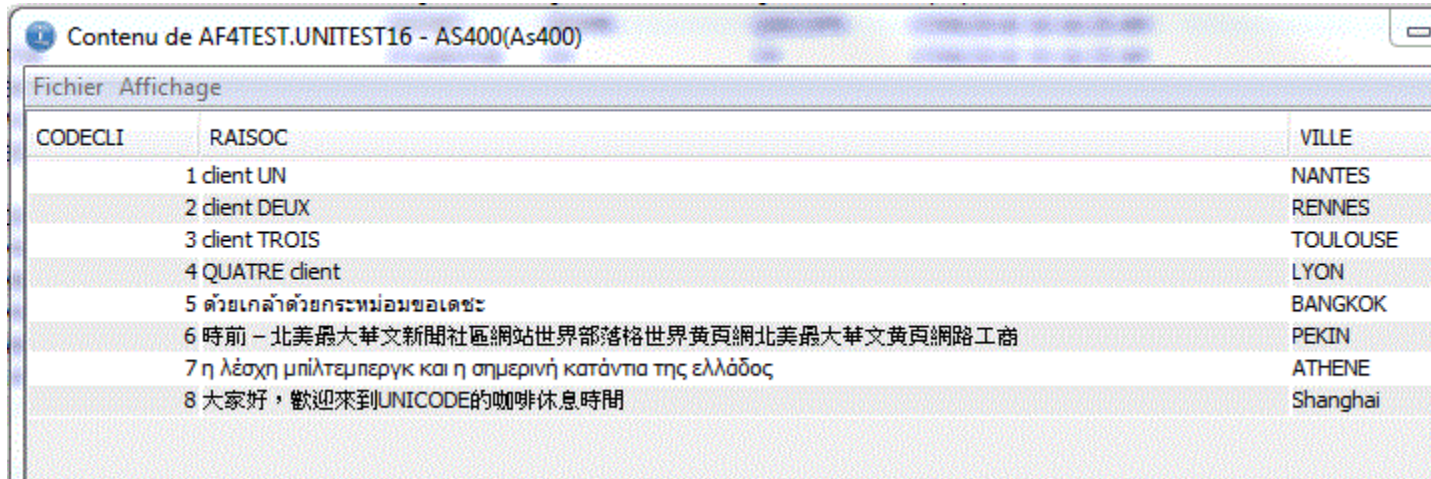


Perspective Database (renvoyer toutes les lignes)

Statut	Résultat1				
	CODECLI	RAISOC	VILLE	DEPCLI	
1	2	client DEUX	... RENNES	... 35	
2	3	МНЕ ВО ФРАНЦИИ, ОТКУ...	MOSCOU	... 31	
3	4	QUATRIÈME CLIENT (ÉÂI) ...	LYON	... 69	
4	5	ด้วยเกล้าด้วยกระหม่อม ขอเดชะ ...	BANGKOK	... 0	
5	6	時 - 北美最大華文新聞社區...	PEKIN	... 99	
6	7	η λέσχη μυλιτεμπεργκ και η ...	ATHENE	... 1	

Affichage et impression

- **ACS sait afficher de l'unicode**
 - Schéma/clic droit sur une table/Données -> Afficher

A screenshot of a database viewer window titled 'Contenu de AF4TEST.UNITEST16 - AS400(As400)'. The window shows a table with three columns: 'CODECLI', 'RAISOC', and 'VILLE'. The table contains eight rows of data, including various client names and their corresponding cities. The text in the 'RAISOC' column is displayed in its original Unicode encoding, showing a mix of Latin, Thai, and Chinese characters.

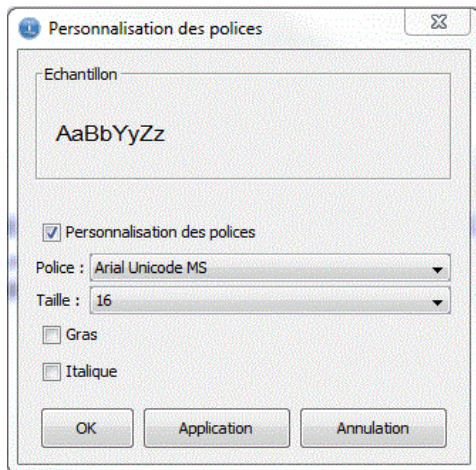
CODECLI	RAISOC	VILLE
1	client UN	NANTES
2	client DEUX	RENNES
3	client TROIS	TOULOUSE
4	QUATRE client	LYON
5	ด้วยเกล้าด้วยกระหม่อมขอเดชะ	BANGKOK
6	時前 – 北美最大華文新聞社區網站世界部落格世界黃頁網北美最大華文黃頁網路工商	PEKIN
7	η λέσχη μιλιτεμπεργκ και η σημερινή κατόνια της ελλάδος	ATHENE
8	大家好，歡迎來到UNICODE的咖啡休息時間	Shanghai

Affichage et impression



- **ACS sait afficher de l'unicode**

- Sous le gestionnaire de script, choisissez d'abord la bonne police (Affichage/Police)



```
1 select * from af4test.unitest16
```

CODECLI	RAISOC
1	client UN
2	client DEUX
3	client TROIS
4	QUATRE client
5	ด้วยเกล้าด้วยกระหม่อมขอเดชะ
6	6 時前 - 北美最大華文新聞社區網站世界部落格世界黃頁網北美最大華文黃頁網路工商
7	7 η λέσχη μπίλτεμπεργκ και η σημερινή καπάνια της Ελλάδος
8	8 大家好，歡迎來到UNICODE的咖啡休息時間



Affichage et impression

- **DSPF, ces derniers connaissent le mot-clé CCSID**
- **CCSID**
 - un CCSID UNICODE
 - 1200 ou 13488
 - *REFC quand vous faites référence
 - Taille
 - une taille d'affichage.
 - *LEN ou non précisé : le double de la taille de stockage
 - *MIN : la taille de la zone (pratique si c'est majoritairement de l'alphabet occidental)

Affichage et impression



- Soit la table suivante

```
CREATE TABLE AF4TEST.UNITEST13 (  
  CODECLI INTEGER NOT NULL ,  
  RAISOC GRAPHIC(80) CCSID 13488 DEFAULT NULL ,  
  VILLE GRAPHIC(50) CCSID 13488 DEFAULT NULL ,  
  DEPCLI DECIMAL(2, 0) DEFAULT NULL )  
RCDFMT UNITEST ;
```

	CODECLI	RAISOC	VILLE	DEPCLI
1	1	client UN	NANTES	44
2	2	client DEUX	RENNES	35
3	3	client TROIS	TOULOUSE	31
4	4	QUATRE client	LYON	69
5	5	ด้วยเกล้าด้วยกระหม่อมขอเดชะ	BANGKOK	0
6	6	時前 - 北美最大華文新聞社區網站世界部落格世界美頁網北美最大華文美頁網路工商	PEKIN	99
7	7	η λέσχη μπύλιτεπεργκ και η σημερινή κατόντια της ελλάδος	ATHENE	1

Affichage et impression



- LE DSPF fait référence

```
UNITESTD.DSPF
Ligne 1      Colonne 1      Remplacement
.....A.....T.Name+++++.....Functions+++++.....Comments+++++
000001      A              R RECORD1      CA03 (03)
000002      A              PAGEDOWN (60)
000003      A              PAGEUP (61)
000004      A              CODECLI      R          0 4 23REFFLD (UNITEST/CODECLI AF4TEST/UNIT-
000005      A              EST13)
000006      A              RAISOC      R          0 6 4REFFLD (UNITEST/RAISOC AF4TEST/UNITE-
000007      A              ST13)
000008      A              VILLE      R          0 9 2REFFLD (UNITEST/VILLE AF4TEST/UNITES-
000009      A              T13)
000010      A              DEPCLI      R          0 12 25REFFLD (UNITEST/DEPCLI AF4TEST/UNITE-
000011      A              ST13)
000012      A              4 5 'Client :'
000013      A              12 4 'Département'
```

Conception Source Aperçu

Propriétés Programme d'invite source Table zone Historique des commandes Table données Liste d'erreurs

AS400: Fichier AF4TEST/UNITEST13 (4 Zones)

Nom	Enregistrement	Type	Longueur	Texte
CODECLI	UNITEST	Binaire	9,0	
RAISOC	UNITEST	Graphique	160	
VILLE	UNITEST	Graphique	100	
DEPCLI	UNITEST	Décimal condensé	2,0	

Affichage et impression



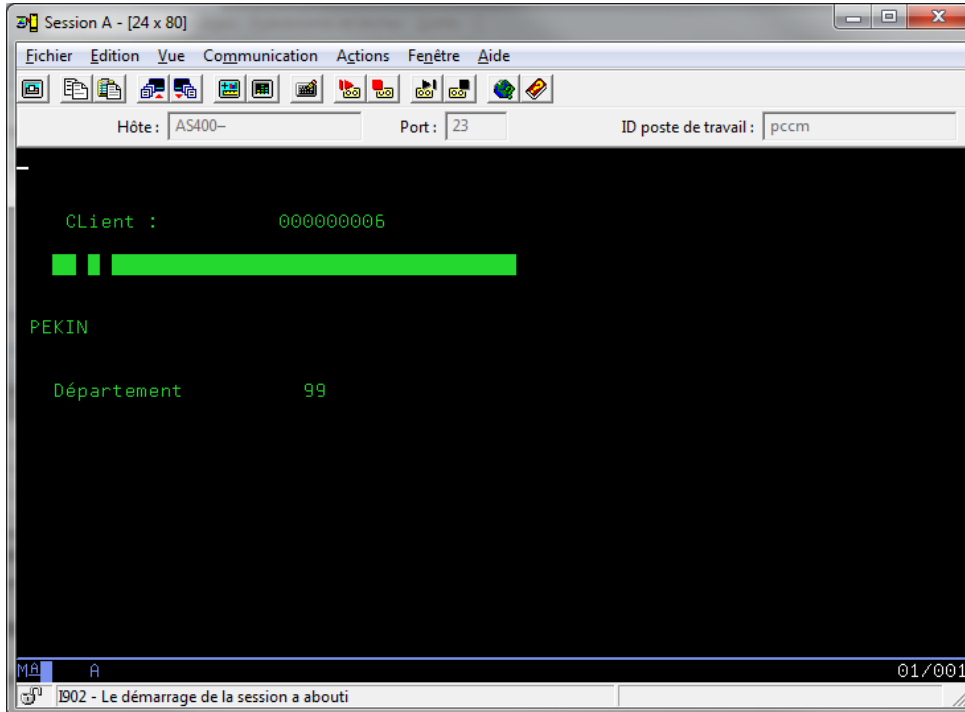
- Ce programme fait un simple affichage

```
UNITSTD.DSPF  UNITSTP.RPGLE X
Ligne 1      Colonne 1      Remplacement
....Ffilename++IPEASFRlen+LKlen+AIDevice+.Keywords+++++++
000100      Funitestd  cf  E          WORKSTN
000101      Funitest13 IF  E          DISK
000102      /free
000103      read unitest;
000104      dow not *in03;
000105          exfmt record1;
000106          if *in60;
000107              read unitest;
000108          ENDIF;
000109      ENDDO;
000110      *inLR = *on;
```

Affichage et impression



- Sur client acces windows



Affichage et impression



- Sur client acces for the web

IBM i Access for Web

Ma page d'accueil			
Mon dossier			
Impression			
Messages			
Travaux			
5250			
• Sessions actives	PEKIN		
• Sessions configurées			
• Démarrage de session			
Base de données			
Fichiers			
Commande			
Téléchargement			
Personnalisation			
Autres			

Client : 000000006

時前 - 北美最大華文新聞社區網站世界部落格世界網頁網北美最大華文網頁網路工商

Département 99

F1	F13
F2	F14
F3	F15
F4	F16
F5	F17
F6	F18
F7	F19
F8	F20
F9	F21
F10	F22
F11	F23
F12	F24

Affichage et impression



- ACS

```
A - unicode
Fichier Edition Affichage Communication Actions Window Aide
Client : 000000005
ด้วยเกล้าด้วยกระหม่อม ขอเด ษ
BANGKOK
Département 00
```

Affichage et impression



- ACS

- Pour cela, vérifiez la présence des options 8, 39 et 43

```
5770SS1      8      Polices compatibles AFP
```

```
5770SS1      39     Composants internationaux pour Unicode
5770SS1      43     Polices supplémentaires
```

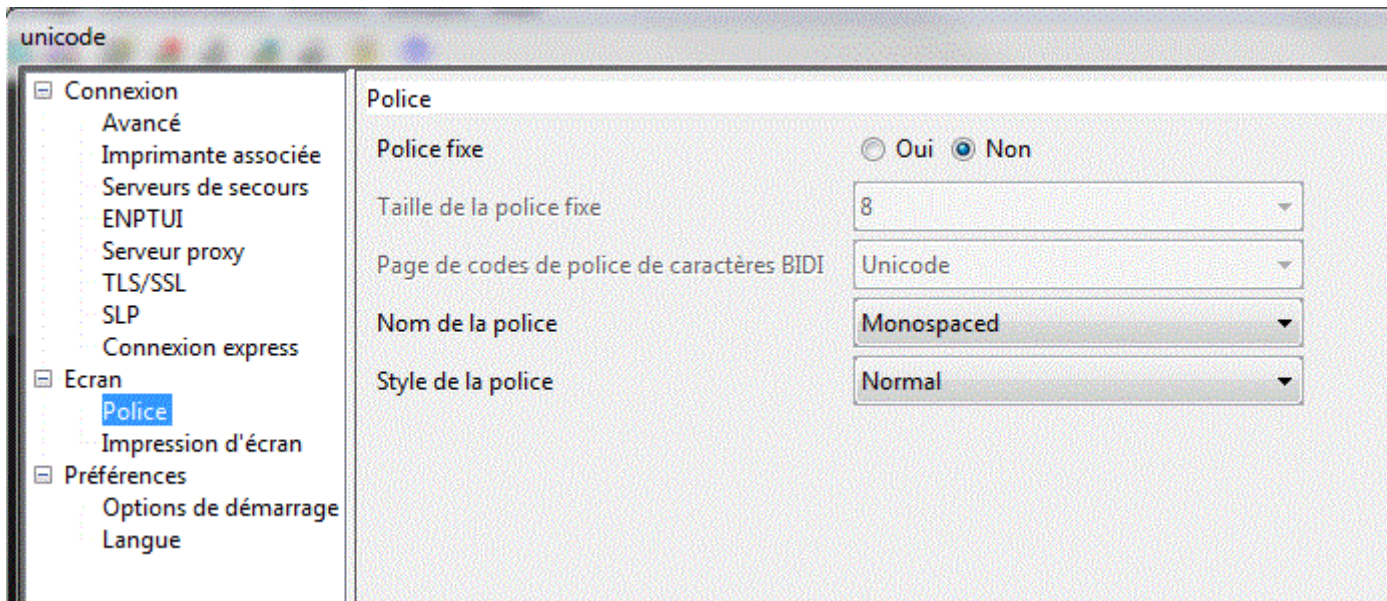
- Définition de la session

Page de codes hôte	1147 Euro - France
Activation du flot de données Unicode	<input checked="" type="radio"/> Oui <input type="radio"/> Non
Activer DBCS dans les zones Unicode	<input checked="" type="radio"/> Oui <input type="radio"/> Non
Connexion automatique	<input checked="" type="radio"/> Oui <input type="radio"/> Non
Reconnexion automatique	<input checked="" type="radio"/> Oui <input type="radio"/> Non

Affichage et impression



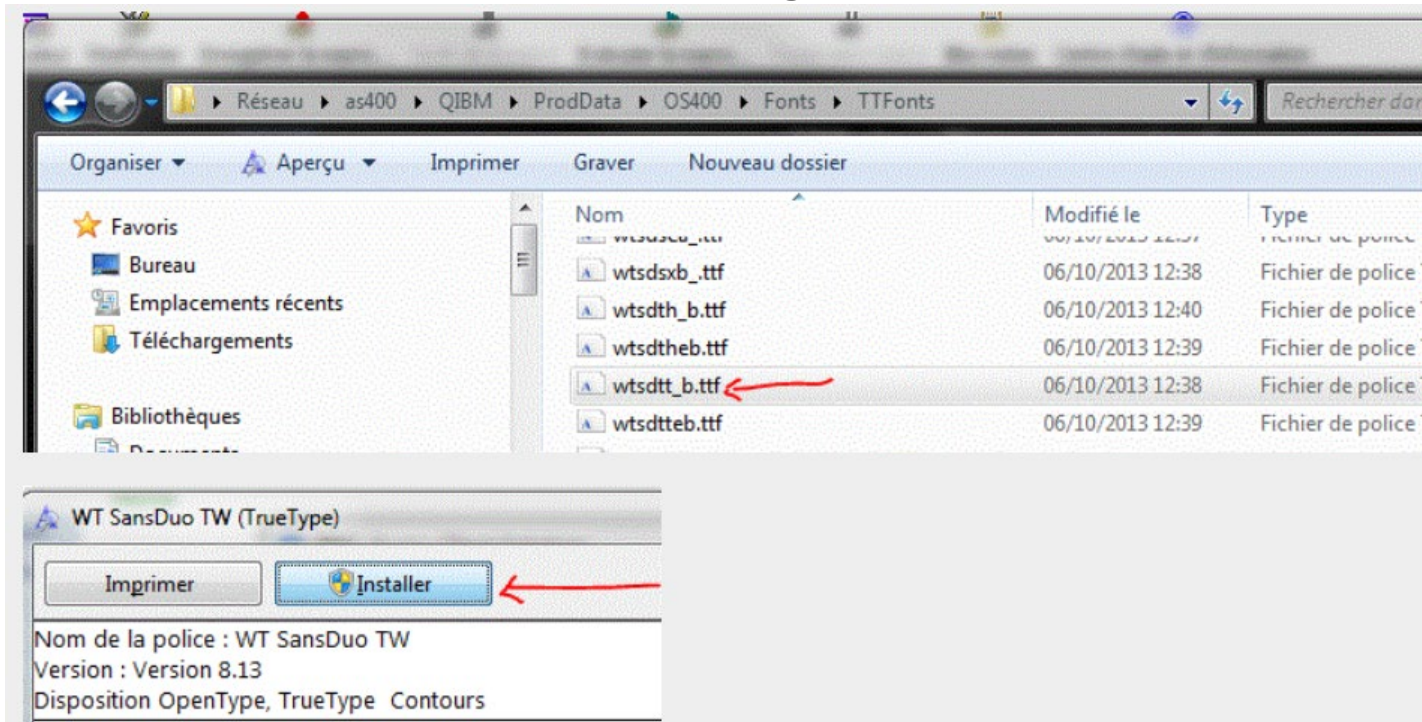
- ACS
 - Définition de la session, suite



Affichage et impression



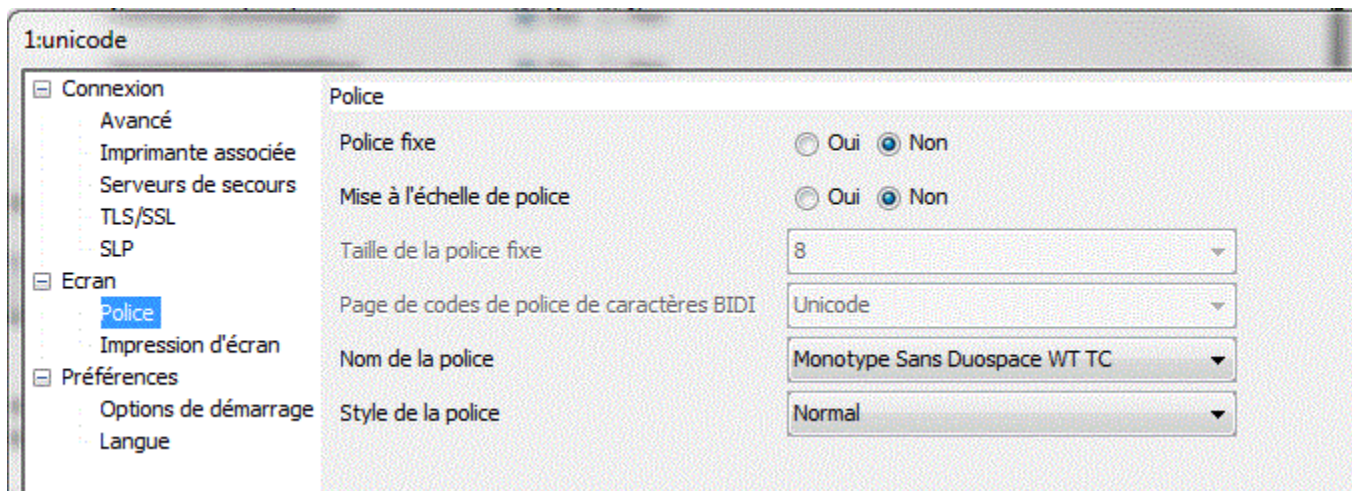
- ACS
 - Pour afficher du chinois, téléchargez depuis l'IBM i



Affichage et impression



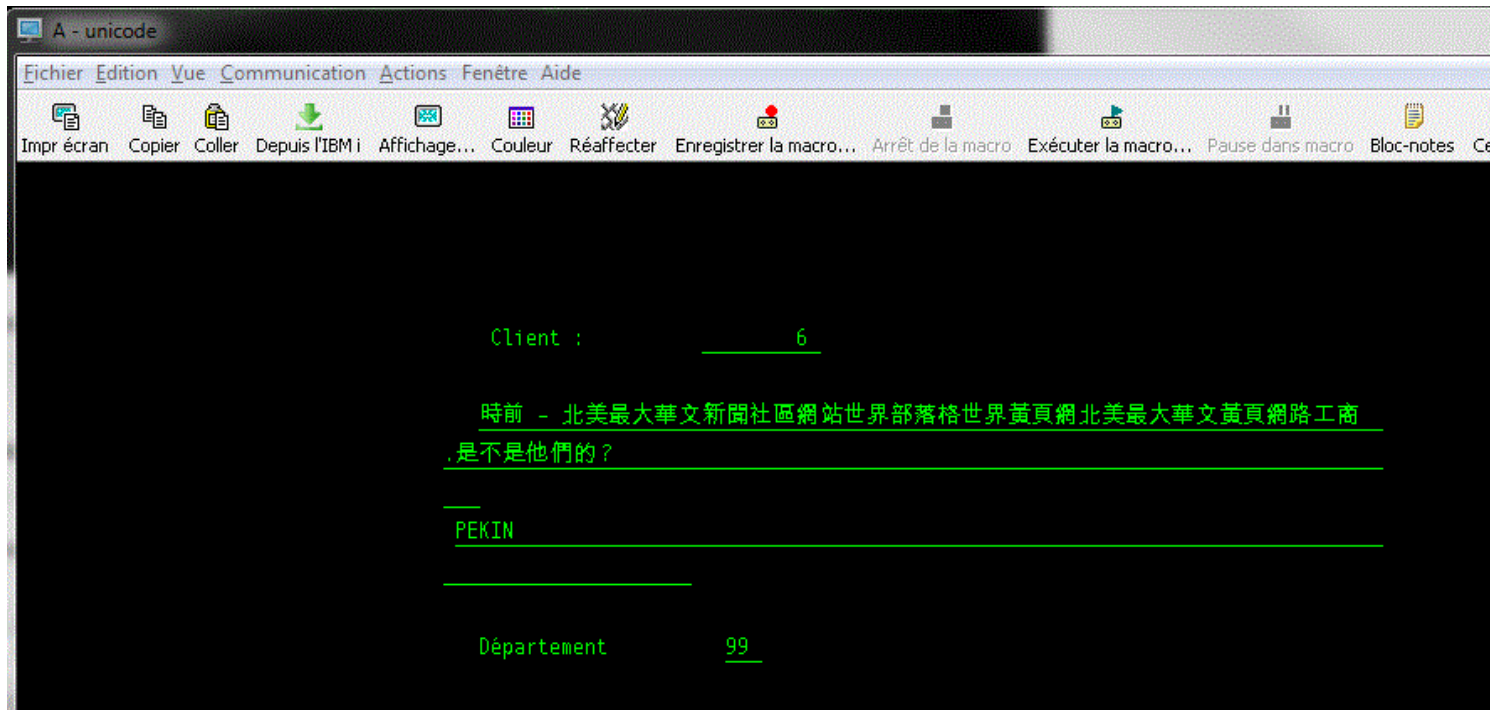
- ACS
 - Définition de la session, suite



Affichage et impression



- ACS





Affichage et impression

- Malgré cela, même sur une session configurée pour Unicode, vous oubliez DFU et QUERY...

```
Option ou commande
==> runqry qryfile(af4test/unitest13)
```

```
Affichage du rapport
                                Largeur du rapport . . . : 288
Première ligne à afficher . . . : _____ Première colonne à afficher . . . : _____
Ligne . . . + . . . 1 . . . + . . . 2 . . . + . . . 3 . . . + . . . 4 . . . + . . . 5 . . . + . . . 6 . . . + . . . 7 . . .
CODECLI RAISOC
000001      2 client DEUX
000002      3 [redacted] ?
000003      4 QUATRIÈME CLIENT (ÉÂÎ)
000004      5 [redacted] [redacted]
000005      6 [redacted] [redacted] ?
000006      7 [redacted] [redacted] [redacted] [redacted]
***** ***** Fin du rapport *****
```

Affichage et impression



- Pour imprimer de l'UNICODE, voyez ce PRTF :

```
UNITESTD.DSPF  UNITESTP.RPGLE  UNITESTI.PRTF X
Ligne 1      Colonne 1      Remplacement
.....A.....T.Name+++++.....Functions+++++Comments+++++
000001      A          R RECORD1      SPACEB(1)
000002      A          43'Liste des clients'
000003      A          R RECORD2      SPACEB(1)
000004      A          CODECLI      R          11SPACEB(5)
000005      A          REFFLD(UNITEST/CODECLI AF4TEST/UNIT-
000006      A          EST13)
000007      A          RAISOC      R          23REFFLD(UNITEST/RAISOC AF4TEST/UNITE-
000008      A          ST13)
000009      A          CCSID(13488 *NOCONVERT)
000011      A          FONTNAME('Monotype Sans Duospace -
000013      A          WT SC' (*POINTSIZ 10))
000014      A          VILLE      R          11SPACEB(1)
000015      A          REFFLD(UNITEST/VILLE AF4TEST/UNITE-
000016      A          T13)
000017      A          DEPCLI      R          121REFFLD(UNITEST/DEPCLI AF4TEST/UNITE-
000018      A          ST13)
000100
```

Rappel : la police 'Monotype Sans Duospace WT SC' est apportée par l'installation de l'option 43

Affichage et impression



- Pour imprimer de l'UNICODE, compilez avec DEVTYPE(*AFPDS)

Pour l'impression nous avons utilisé une Imprimante Réseau avec MFRTYPMDL(*HPDBCS)

scan de la page



Liste des clients	
00000002 RENNES	client DEUX
00000003 TOULOUSE	client TROIS
00000004 LYON	QUATRE client
00000005 BANGKOK	ด้วยเกล้าด้วยกระหม่อมขอเดชะ
00000006 PEKIN	時前 - 北美最大華文新聞社區網站世界部落格世界黃頁網北美最大華文黃頁網路工商
00000007 ATHENE	η λέσχη μιλιτεμεργκ και η σημερνή καπάντια της ελλάδος

Exportation



- CPYTOIMPF

Copier dans fich importation (CPYTOIMPF)

Indiquez vos choix, puis appuyez sur ENTREE.

```
Fichier d'origine: FROMFILE
  Fichier . . . . .
  Biblio . . . . .
  Membre . . . . .
Fichier BD de destination: TOFILE
  Fichier . . . . .
  Biblio . . . . .
  Membre . . . . .
Fichier STREAM de destination . TOSTMF

Remplacement ou ajout enregs . . MBROPT
CCSID d'origine . . . . . FROMCCSID
CCSID fichier STREAM . . . . . STMFCCSID
Droit d'accès fichier STREAM . . STMFAUT
Délimiteur d'enregistrement . . RCDDL
Format enr fichier importation . DTAFMT
```

```
> UNITEST16
> AF4TEST
*FIRST
_____
*LIBL
*FIRST
> '/formation/unicode.csv'
_____
*ADD
*FILE
> 1200
*DFT
*CRLF
*DLM
```


Langage de contrôle



- De manière générale, la documentation indique que le CL admet l'Unicode depuis la 7.3. (bien partiellement à notre avis)
- https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_73/rbam6/rbam6unicode.htm
- En effet :
- Les variables CL ne peuvent pas être en Unicode, seules les constantes sont admises
 - Et encore, voyez ci-dessous les limites de RDI avec un QCLSRC en UTF-8, qui le seul codage Unicode admis pour les sources
- Les paramètres des commandes peuvent être en Unicode (peu nombreuses, mais vous pouvez en créer))

```
PARM KWD (&MOTCLE)
      TYPE (*CHAR) LEN (200)
      CCSID (*UTF16)
      PROMPT ('Paramètre en UNICODE')
```

Langage de contrôle



- Si vous avez besoin, l'Api iconv permet de convertir, en CL, vers UNICODE
- QCMD ne sait **pas** traiter de l'UNICODE
- QCMD EXC non plus, utilisez **QCAPCMD**
- **La ligne de commande RDI est « unicode compatible »**

Voyez

https://github.com/FrenchIBMi/exemples/tree/master/CL_UNICODE

A screenshot of an AS400 command window. The window title is 'AS400'. The command history shows three commands: 'ADDLIB BDVIN1 *LAST', 'ADDLIB FORMATION1 *LAST', and 'MKDIR DIR('/formation/test/RDI/DBE каранор')'. The output for each command indicates that the library or directory was successfully added or created. The command 'MKDIR DIR('/formation/test/RDI/三個目錄)' is currently being executed, and its output shows that the directory was created.

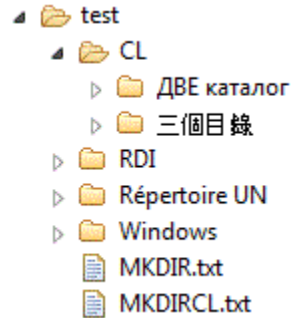
```
AS400
ADDLIB BDVIN1 *LAST
Bibliothèque BDVIN1 ajoutée à la liste des bibliothèques.
Cause . . . . . : Si vous avez utilisé la commande ADDLIB (Ajouter un poste à la liste des bibliothèques de la liste des bibliothèques.
ADDLIB FORMATION1 *LAST
Bibliothèque FORMATION1 ajoutée à la liste des bibliothèques.
Cause . . . . . : Si vous avez utilisé la commande ADDLIB (Ajouter un poste à la liste des bibliothèques de la liste des bibliothèques.
MKDIR DIR('/formation/test/RDI/DBE каранор')
Répertoire créé.
Cause . . . . . : Le répertoire /formation/test/RDI/DBE каранор a été créé.
MKDIR DIR('/formation/test/RDI/三個目錄')
Répertoire créé.
Cause . . . . . : Le répertoire /formation/test/RDI/三個目錄 a été créé.

Commande Normale MKDIR DIR('/formation/test/RDI/三個目錄')
```

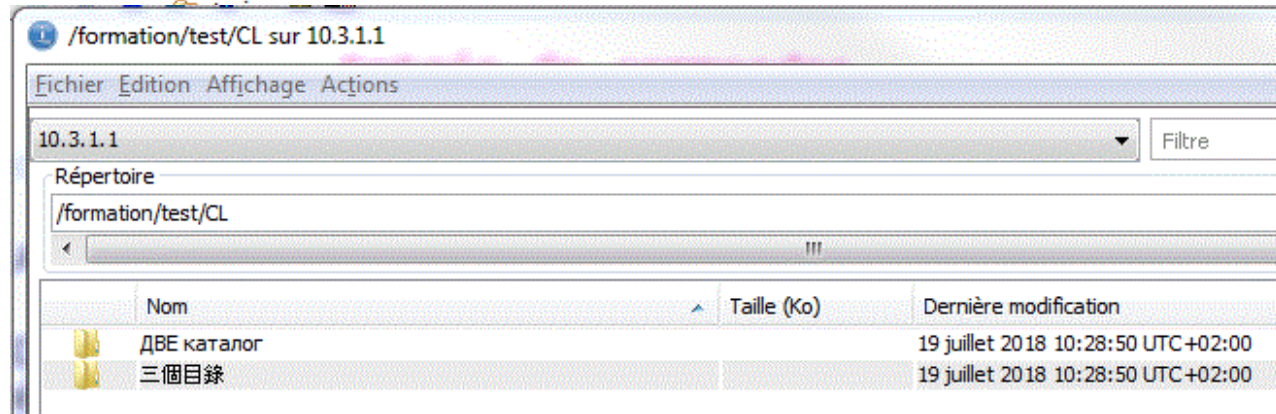
Langage de contrôle



- Résultat
 - Sous RDI



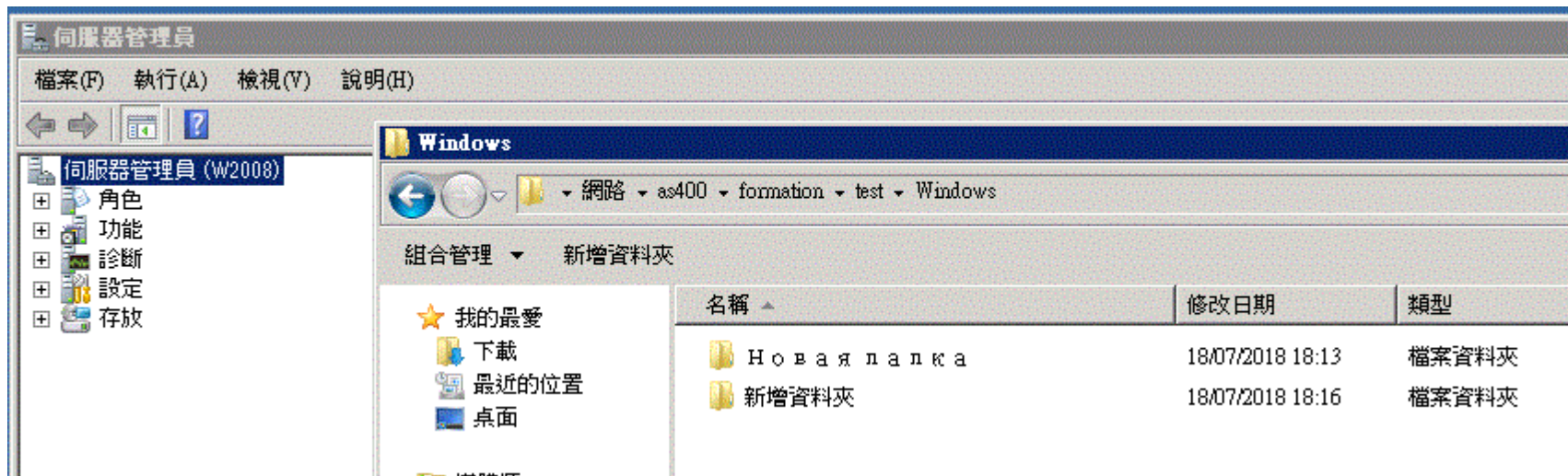
- Avec ACS



Langage de contrôle



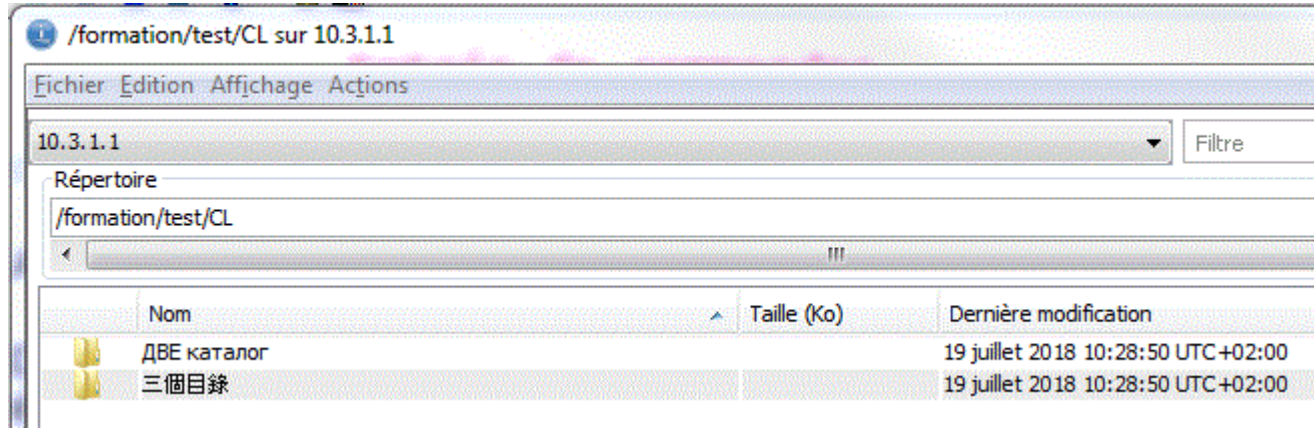
- Nous avons aussi installé les langues Russe et Chinoise sur un Windows et demandé la création d'un répertoire



Langage de contrôle



- Résultat
- ACS



- Navigator for i



- RPG reconnaît les champs **Graphic**

DCL-S nom-variable, type *mots-clés* :

<i>Type</i>	<i>remarque</i>	<i>équivalent D</i>	<i>paramètres</i>
TIMESTAMP	horodatage	Z	
UCS2 <-	Unicode	C	UCS2(lg)
UNS	binaire non signé	U	Uns(lg)
VARCHAR	Alphanumérique à taille variable	A + VARYING	Varchar(lg)
VARGRAPH	DBCS à taille variable	G + VARYING	Vargraph(lg)
VARUCS2 <-	Unicode à taille variable	C + VARYING	Varucs2(lg)

- En 7.2 avec le mot-clé CCSID,
une variable peut avoir une codification spécifique :
 - CCSID(*UTF8) , donnée unicode
 - CCSID(819) , données ASCII standard
 - CCSID(37), EBCDIC Américain (rappel : 297 pour la France)
 - CCSID(*HEX), pas de gestion du CCSID pour cette zone
- L'adaptation des constantes au CCSID de la variable est automatique
plus besoin des fonctions %UCS2()

- En 7.2 avec le mot-clé CCSID,

```
000100 **free
000101 ctl-opt ccsid(*UCS2 : 1200);
000102 dcl-s zone ucs2(10);
000103
000104 zone = 'bonjour';
000105 if %subst(zone:1:1) = 'b';
000106     dsply 'OK';
000107 ENDIF;
000108 *INLR = *ON;
```

Moniteurs

ZONE : 0xD3E5E9963D002530 <Hexadécimal brut>

Adresse	0 - 3	4 - 7
D3E5E9963D002530	0062006F	006E006A
D3E5E9963D002540	00700070	00000000

- Les fonctions %Date, %Dec ne transforment pas directement de l'UNICODE, mais en passant d'abord par %CHAR, cela fonctionne

```
1 ctl-opt ccsid(*UCS2 : 1200);
2 dcl-s zone ucs2(10);
3 dcl-s zonedate date;
4
5 zone = '2018-01-31';
6 zonedate = %date(%char(zone));
7
8 dcl-s zone ucs2(10);
9 dcl-s zonedec packed(7:2);
10 dcl-c l %len(zonedec);
11 dcl-c d %decpos(zonedec);
12
13 zone = '- 123,45';
14 zonedec = %dec(%char(zone):l:d) ;
15 if zonedec < 0;
16     dsply 'négatif';
17 ENDIF;
```



- Et il y a des nouvelles directives de compilation en 7.2

/SET permet de définir le CCSID (ou le format date/heure) à partir de cette ligne

- /SET CCSID(*CHAR : n°-de-ccsid)
 - /SET CCSID(*GRAPH : n°-de-ccsid)
-
- /RESTORE, pour annuler le précédent /SET

RPG et services Web



- Ce programme va être exposé en tant que web service REST

```

-
**free
ctl-opt alwnull(*usrctl) pgminfo(*pcml:*module) ;
dcl-f unitest13 disk;
dcl-pi *n;
  code like(codecli);
  nom like(raisoc);
End-Pi;
chain code unitest13;
if %found;
  nom = raisoc;
else;
  nom = '** non trouvé **';
EndIf;
*inlr = *on;
```

RPG et services Web



- Définition des paramètres

Export procedures: ?

Select	Procedure name/Parameter name	Usage	Data type
<input checked="" type="checkbox"/>	▼ UNITESTWS		
	CODE	input ▼	packed
	NOM	output ▼	char

Select All Deselect All Expand All Collapse All

- Fichier swagger

```
▼ UNITESTWSResult:  
  type: "object"  
  ▼ properties:  
    ▼ NOM:  
      type: "string"  
      maxLength: 80
```

RPG et services Web



•Résultat

as400:10025/web/service X

← → ↻ ⓘ Non sécurisé | as400:10025/web/services/UNITESTWS/5 ☆

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<UNITESTWSResult>
  <NOM>時 - 北美最大華文新聞社區網站世界部落格世界黃頁網北美最大華文黃頁網路工商 .是不是他們的? </NOM>
</UNITESTWSResult>
```

