

À plus de 60 ans, le langage COBOL est encore utilisé par des États américains,


Les laissant face à un manque de programmeurs et des coûts de développement énormes

Le 4 mars 2021 à 03:37, par Michael Guilloux 14 commentaires

     75 PARTAGES

 17

 0

 Créé en 1959, le langage de programmation COBOL est encore utilisé dans de nombreuses organisations. La conséquence est que celles-ci souffrent d'un manque de programmeurs. Elles doivent également fournir de grands efforts de reprogrammation, même pour les plus petits changements, comme c'est le cas dans certains États américains avec leurs systèmes informatiques de chômage.

Avec la crise sanitaire de la COVID-19, les États-Unis ont en effet mis en place un plan de relance pour soutenir les personnes ayant perdu leur emploi durant la pandémie. Ce plan nécessitait pour chaque État d'implémenter des changements dans leurs systèmes informatiques de chômage. Plus un État prenait du temps, plus les personnes qui ont été mises au chômage à cause du coronavirus devront attendre avant de bénéficier de l'aide du gouvernement.

Certains États ont eu des difficultés particulières à implémenter les changements requis simplement à cause du langage dans lequel leurs systèmes sont écrits : COBOL. Eh oui, la soixantaine révolue, COBOL est encore utilisé par certains États pour leurs systèmes de chômage. C'est le cas notamment de l'Iowa, le New Jersey, le Kansas et l'Oklahoma.



À part l'Iowa qui avait suffisamment de développeurs COBOL, les autres États ont été confrontés à des difficultés importantes. Lorsqu'ils ont eu besoin de programmeurs pour mettre à jour leur « vieux code COBOL que personne n'a touché en 20 ans », cela est devenu un problème. « Ils ont réalisé que tous leurs programmeurs COBOL avaient pris leur retraite ou étaient décédés », a déclaré Dennis Brylow, professeur d'informatique à l'Université Marquette. Dans le même temps, de nombreuses universités n'enseignent plus le langage COBOL de manière approfondie, voire pas du tout, ce qui se traduit par très peu de nouveaux programmeurs COBOL.

Et même quand il y a les hommes qu'il faut, les problèmes avec COBOL ne sont pas encore terminés. Certains États ont essayé de moderniser leurs systèmes et se sont heurtés à des obstacles. Il s'agissait précisément de coûts de développement et de retards particulièrement élevés. Comme l'explique Brylow, le passage complet d'un système COBOL à un autre langage de programmation est loin d'être une solution parfaite. « Quand vous voyez quelque chose qui n'a besoin que de quelques ajustements et que c'est un gros système qu'ils ont passé des années à construire, c'est une dépense énorme et c'est en fait assez dangereux d'essayer de le réimplémenter totalement dans un nouveau langage de programmation », a-t-il déclaré.

S'il est vieux et dit dépassé, COBOL est étonnamment encore présent, notamment dans le secteur des banques et services financiers où il serait d'ailleurs le plus utilisé. Qu'est-ce pourrait donc expliquer cela ?

Et vous ?


- ➔ Pourquoi COBOL est-il encore populaire aujourd'hui dans le secteur des finances ?
- ➔ Qu'est-ce qui empêche les entreprises de ce secteur de passer à des technologies plus modernes ?
- ➔ COBOL a-t-il encore de beaux jours devant lui ?

Voir aussi :





- ➔ Êtes-vous un développeur COBOL ? Si oui, il pourrait y avoir encore des opportunités pour vous dans le domaine de la finance
- ➔ Micro Focus annonce la sortie de Visual COBOL pour Visual Studio 2017 qui offre aux développeurs COBOL la possibilité de coder avec l'EDI
- ➔ Jean E. Sammet, une informaticienne qui a participé au développement de COBOL, est morte à l'âge de 89 ans

14 COMMENTAIRES

 Commenter

 Signaler un problème

Les mieux notés

<div>i5evangelist</div> <div></div> <div>Membre éclairé</div> <div>Le 04/03/2021 à 5:19</div>	<div>« Pourquoi COBOL est-il encore populaire aujourd'hui dans le secteur des finances ? »</div> <div>« Le Cobol est un langage adapté à la gestion, vous voulez le remplacer par quoi ? du python ? 😊 »</div> <div>« Qu'est-ce qui empêche les entreprises de ce secteur de passer à des technologies plus modernes ? »</div> <div>« CA veut dire quoi des technologies plus modernes ? Un technologie qui sera obsolète avant même que le projet de réécriture soit fini ? (reprenez les news de developpez.com d'il y a 10 et vous comprendrez) »</div> <div>« COBOL a-t-il encore de beaux jours devant lui ? »</div> <div>« Disons que pour ce genre de situation, bien sûr »</div> <div><div> 7</div><div> 0</div></div>
<div>el_slapper</div> <div></div> <div>Expert éminent sénior</div> <div>Le 04/03/2021 à 16:06</div>	<div>« Le problème n'est pas le langage, mais les pratiques de développement qui avaient cours durant les précédentes décennies. Quand on doit se forcer la reprise de ces programmes antédiluviens, peu importe que ce soit du Cobol, du PL/I ou du RPG, le nombre de WTF par minute explose. Je conseillerai aux boîtes qui reposent sur du Cobol non pas de changer de langage, mais de nettoyer leur code et de mieux l'architecturer. »</div> <div>yahiko</div> <div>« Souvent vrai. Pas toujours, mais les codes des années 1970 étaient très souvent horribles, j'en ai maintenu quelques uns, et il m'est même arrivé de les refondre. Une des contraintes, c'était la place mémoire. A tel point que les noms de variables ou de labals (pour les GO TO...) étaient souvent sur 2-3 caractères maximum. »</div> <div>code de 1972 :</div> <div>Code :</div> <div><pre>1 A12. 2 MOVE 1 TO YY 3 A122. 4 IF XX(YY) = 0 5 GO TO A121. 6 MOVE XX(YY) TO PP(YY) 7 ADD 1 TO YY 8 GO TO A122. 9 A121.</pre></div> <div>Sélectionner tout</div>

Confidentialité

parait que vous savez développer en COBOL...

>>

**

➔ [Le langage de programmation Cobol fête ses 60 ans](#)

peut-il encore tenir longtemps face à la concurrence des nouveaux langages ?

👍 2

🗨️ 0

abriotde

Membre chevronné

Le 04/03/2021 à 16:49

<<

i5evangelist

>>

Aujourd'hui, tout est concentré sur le partage et l'exposition des données et la modernisation des interfaces.
Dans le domaine bancaire, je ne suis pas certain que l'exposition des données soit à l'ordre du jour 😊

Dans le domaine bancaire comme ailleurs, il faut exposer les données au client (dans son smartphone), au banquier (Sur son tableur), au comptable, au fisc, aux autres banques. Aujourd'hui les banques sont hyper-connectées.
Et si on expose les données, cela ne veut pas dire qu'il n'y a pas de backend.

👍 1

🗨️ 0

TotoParis

Membre expérimenté

Le 05/03/2021 à 21:49

<<

Andarus

>>

Oui, tout à fait, dans de grosse entités, par de grosses SSII.

<<

el_slapper

>>

Ne pas oublier en plus que dans les années 1970, l'écriture sur bordereau papier et transcription en cartes perforées, pour compilation la nuit et récupération du listing le lendemain matin étaient encore en vigueur...Et aussi, l'enseignement de la programmation était très peu poussé.

Ce qui manque à COBOL, ce sont de vrais outils comme ECLIPSE, mais qui ne coûtent pas 1 bras comme RDZ d'IBM (1 bras par licence).
Un débbugger interactif comme XPEDITER, c'est génial, sous TSO ou en ayant lancé son JCL et avec interception sous TSO.
FILE-AID (fichiers, DB2) est génial là aussi.
Le langage est simple à apprendre. Mais il faut se méfier des options de compilation, très farceuses pour les zones numériques.
L'interface TSO est très spartiate, mais pas pire que dans le monde UNIX, voire mieux avec les écrans ISPF (et les puissants langages de script comme CLIST et REXX).
Par contre, merci les "GO TO" : une plaie connue depuis 40 ans au moins, mais toujours utilisables.
Pour définir une structure de fichier, ou de table DB2, il n'y a pas mieux, à part PL/1, un peu comme les RECORD en Pascal, voir https://pascal.developpez.com/cours/.../?page=page_12.
Il y a énormément de fonctions avec Language Environment. Mais il manque comme dans un langage plus moderne la possibilité de déclarer ses propres fonctions dans un programme, là encore, comme en Pascal : https://www.tutorialspoint.com/pasca..._functions.htm, <https://www.tutorialspoint.com/pasca...procedures.htm>.
Car sinon la WORKING-STORAGE SECTION ressemble a un vaste grenier bien bordélique un peu trop souvent...et PROCEDURE DIVISION alors là, lol mais lol quoi...

👍 1

🗨️ 0

📝 Commenter

🚩 Signaler un problème