



VERYANT E isCOBOL

**NA VERYANT, NOS
PREOCUPAMOS
EM TORNAR SUA
VIDA MAIS FÁCIL.**



Esta edição destaca 2020R2, com muitas maneiras de trabalhar de forma mais inteligente, incluindo

- Um Profiler mais acessível,
- Um Debugger otimizado
- Um apelo para usar a IDE isCOBOL.



**Nossos pensamentos
estão com todos os
que foram afetados
pelo coronavírus.**

veryant

NEWS

NESTA EDIÇÃO

1. Notícias de lançamento 2020R2
2. isCOBOL Profiler - Destaques de Documentação - Você já viu isso?
3. COBOL e nossas Universidades
4. Tchau! iscobol.debug.code_prex - Russell Kirsch
5. Você não utiliza a IDE? É fácil começar - Hamburger Menu - Modernize os ícones de seus aplicativos
6. Última página

NOVIDADES

2020R2 chegou

Veryant tem o prazer de anunciar o lançamento mais recente do isCOBOL™ Evolve, isCOBOL Evolve 2020 Versão 2.



Cada lançamento é empolgante para nós, porque podemos mostrar aos nossos clientes que os ouvimos e que estamos trabalhando muito para tornar nossos produtos mais poderosos, fáceis e relevantes para as tendências atuais. Este lançamento não é exceção.

2020R2 torna mais acessível a caracterização do seu aplicativo, tanto para executar quanto para ajustar as suas necessidades. A depuração é mais simples e transparente - consulte o artigo intitulado "Dizendo tchau ao iscobol.debug.code_prex" abaixo.

A IDE tem mais recursos integrados, então você não precisa deixá-la para criação de profiling, code coverage ou unit testing. Nossos clientes sempre perguntaram sobre a capacidade de poder modificar suas variáveis fora dos editores gráficos e, nesta versão, fizemos isso. Não usa nossa IDE? Veja as razões pelas quais você deveria utilizar lendo o artigo de Davide Spizzi, Veryant's Support Engineering Manager.

Mudanças gráficas como hamburger menu, criação de bitmaps usando glyph symbols, um novo web browser class e maior performance de compilação foram adicionados na 2020R2. Saiba mais e veja o vídeo de demonstração em nosso [canal no YouTube](#).



DESTAQUES DA DOCUMENTAÇÃO



Performance Tuning

Organizamos todas as nossas sugestões de melhoria de desempenho em um só lugar para tornar mais fácil para você ajustar a velocidade de seu aplicativo. Nos anexos da documentação do isCOBOL Evolve há um livro chamado [Ajuste de Desempenho](#). Existem três capítulos, “Diretrizes para compilação mais rápida”, “Diretrizes para melhor desempenho em tempo de execução” e “Criação de profiler de programas COBOL”.

Você encontrará sugestões para uma compilação mais rápida da linha de comando e da IDE, otimizações durante a compilação e execução de seu aplicativo e sugestões para Thin Client, acesso a dados e melhoria de desempenho de impressão.

O capítulo Criação de Profiling tem algumas mudanças significativas na 2020R2 - leia mais sobre isso nesta página.

Conheça o novo isCOBOL Profiler

isCOBOL Profile Report

Executed: July 1, 2020 12:55:13 PM

Elapsed: 3.144s; Evaluated: 2.919s; Overhead1: 41; Overhead2: 43

Warning: some programs are compiled in debug mode, this affects performances

Search:

[View Program table](#)

Program	Paragraph	Self %	Seconds	Count
IO_INDEXED	DELETE_FILE1_TEST	28.38%	0.828	1
IO_INDEXED	UPDATE_FILE1_TEST	15.94%	0.465	1
IO_INDEXED	LOAD_FILE1_TEST	11.33%	0.331	1
IO_LINSEQUENTIAL	LOAD_FILE1_TEST	6.81%	0.199	1
IO_INDEXED	READ_FILE1_TEST	6.59%	0.192	1
IO_RELATIVE	DELETE_FILE1_TEST	4.65%	0.136	1
IO_RELATIVE	UPDATE_FILE1_TEST	4.37%	0.128	1
IO_RELATIVE	LOAD_FILE1_TEST	3.62%	0.106	1
IO_SEQUENTIAL	LOAD_FILE1_TEST	3.61%	0.105	1
IO_LINSEQUENTIAL	READ_FILE1_TEST	3.05%	0.089	1
IO_PERFORMANCE	MAIN_LOGIC	2.88%	0.084	1
IO_RELATIVE	READ_FILE1_TEST	1.67%	0.049	1
IO_RELATIVE	START_TIMER	1.58%	0.046	4
IO_SEQUENTIAL	READ_FILE1_TEST	1.47%	0.043	1
IO_INDEXED	START_TIMER	1.27%	0.037	4
IO_INDEXED	MAIN_LOGIC	1.04%	0.03	1
IO_SEQUENTIAL	START_TIMER	0.89%	0.026	2
IO_LINSEQUENTIAL	START_TIMER	0.54%	0.016	2
IO_RELATIVE	MAIN_LOGIC	0.13%	0.004	1
IO_LINSEQUENTIAL	MAIN_LOGIC	0.08%	0.002	1
IO_SEQUENTIAL	MAIN_LOGIC	0.07%	0.002	1
IO_INDEXED	STOP_TIMER	0.02%	0.001	4
IO_RELATIVE	STOP_TIMER	0.01%	0	4
IO_SEQUENTIAL	STOP_TIMER	0.00%	0	2
IO_LINSEQUENTIAL	STOP_TIMER	0.00%	0	2
Totals:		100.00%	2.919	

[View Program table](#)

isCOBOL Profile Report

Executed: July 1, 2020 12:55:13 PM

Elapsed: 3.144s; Evaluated: 2.919s; Overhead1: 41; Overhead2: 43

Warning: some programs are compiled in debug mode, this affects performances

Search:

[View Paragraph table](#)

Program	Self %	Seconds	Count
IO_INDEXED	64.58%	1.885	1
IO_RELATIVE	16.02%	0.468	1
IO_LINSEQUENTIAL	10.48%	0.306	1
IO_SEQUENTIAL	6.04%	0.176	1
IO_PERFORMANCE	2.88%	0.084	1
Totals:	100.00%	2.919	

[View Paragraph table](#)

isCOBOL Profile Report

Executed: July 1, 2020 12:55:13 PM

Elapsed: 3.144s; Evaluated: 2.919s; Overhead1: 41; Overhead2: 43

Warning: some programs are compiled in debug mode, this affects performances

Search:

[View Program table](#)

Program	Paragraph	Self %	Seconds	Count
IO_INDEXED	DELETE_FILE1_TEST	28.38%	0.828	1
IO_INDEXED	UPDATE_FILE1_TEST	15.94%	0.465	1
IO_INDEXED	LOAD_FILE1_TEST	11.33%	0.331	1
IO_INDEXED	READ_FILE1_TEST	6.59%	0.192	1
IO_INDEXED	START_TIMER	1.27%	0.037	4
IO_INDEXED	MAIN_LOGIC	1.04%	0.03	1
IO_INDEXED	STOP_TIMER	0.02%	0.001	4
Totals:		64.57%	1.884	

[View Program table](#)

A 2020R2 inclui melhorias no isCOBOL Profiler, tornando o relatório mais simples de ser criado e visualizado. A execução de seus programas com a opção “-profiler” criará um relatório HTML mostrando o tempo e a porcentagem do tempo total para cada programa, bem como cada parágrafo nos programas. Novas opções foram adicionadas para ajudá-lo a obter um relatório melhor, incluindo: Biblioteca C\$PROFILER - para ligar e desligar o profiler programaticamente, especialmente

em torno de instruções ACCEPT. Isso remove o tempo de ação do usuário. “Iscobol.profiler.excludes” e “iscobol.profiler.includes” - Para restringir o classes profiling. Isso é útil se sua IU for separada ou se você tiver identificado um problema para um programa. O relatório mostra um aviso se os programas profiled foram compilados no debug. Você deve apenas criar profiler de programas NÃO compilados para depuração.

Você viu isso?

Novos vídeos do YouTube:

[New Features of isCOBOL Version 2020R1](#)

[isCOBOL for the Technologist](#)

[Modernization Methods with isCOBOL](#)

[isCOBOL and REST Services Demo](#)

[Mainframe Rehosting with HTWC](#)

[isCOBOL e zOS](#)

Novos artigos da base de conhecimento:

[Como faço para indicar meu isCOBOL instalado para um local Java diferente?](#)

[Como estender o comprimento da ÁREA B no código-fonte de um programa](#)

[Como executar programas em lote em um servidor Linux e compilá-los e depurá-los via IDE](#)

[Como posso controlar ou restringir o acesso aos meus aplicativos WebClient e ver o nome de usuário exibido no console do Web-Client](#)

[Como copiar dados da área de transferência](#)



Evolution without revolution

COBOL e nossas Universidades



Marilyn Prince, engenheira de vendas para a América do Norte, fala sobre a necessidade de instrução sobre COBOL em nossas universidades, como manter os alunos interessados e o que a Veryant está fazendo para ajudar

A escassez de programadores COBOL tem estado muito nos noticiários ultimamente, pois estes desenvolvedores estão envelhecendo e perdendo a força do trabalho. COBOL é ensinado em pouquíssimas universidades, e frequentemente, como eletiva. Para resolver o problema, precisamos ver por que há uma grande lacuna entre a necessidade de programadores COBOL e a falta de prioridade para treiná-los.

Pode ser porque se esperava que a maioria dos programas COBOL fossem substituídos até o ano 2000 (levando ao problema Y2K). Ou talvez enquanto os programas COBOL “ronronam” alegremente, a administração complacente com a necessidade de mudança - a velha máxima “se não está quebrado, não conserte”. Talvez as universidades - empresas com fins lucrativos sujeitas aos caprichos de seus alunos - tenham considerado o COBOL algo fora de moda entre seus alunos, seguindo o caminho das calças boca de sino e da Beatlemania.

A resposta é provavelmente uma combinação de todas as três. Mas provamos que o COBOL ainda está aqui e provavelmente veio para ficar de uma forma ou de outra. Precisamos não apenas de mais jovens programadores COBOL, mas também de programadores COBOL que possam trazer nossos aplicativos para o século 21.

Os programadores COBOL não podem mais se basear apenas conhecendo o COBOL. Por exemplo, COBOL e Java estão sendo usados juntos com mais frequência, em parte porque os aprimoramentos recentes da IBM permitem que COBOL e Java sejam executados juntos em mainframes. As duas linguagens se encaixam bem, é por isso que isCOBOL é escrito em Java e compila o código COBOL para classes Java.

Os recursos do COBOL como serviços da Web REST ou SOAP significam que o COBOL pode continuar a funcionar facilmente como uma linguagem de processamento de dados, enquanto as linguagens mais recentes podem fornecer interfaces mais chamativas que exibem facilmente os dados em telas com gráficos, tabelas e painéis. Mesmo um aplicativo estritamente COBOL pode ser aprimorado com Programação OO e a inclusão de JavaBeans, por exemplo. É comum ver implementação COBOL com outras linguagens ou ferramentas combinadas.

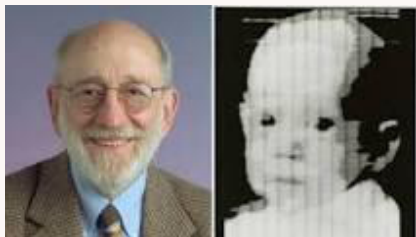
Um comentarista do Hacker News observou que um programador COBOL passa sua carreira “fazendo trabalho de manutenção em vez de qualquer desenvolvimento greenfield”. Isso pode ser verdade em alguns casos, mas para avançar para o futuro, os programas COBOL terão que evoluir, e o novo desenvolvimento é uma parte disso - seja na forma de telas gráficas COBOL, programação COBOL orientada a objetos ou particionamento de COBOL em serviços web.

Mas como fazer com que os alunos se interessem o suficiente para pagar as universidades para ensinar COBOL? Novos alunos estão limitados pelo ambiente de desenvolvimento de “tela verde”, mas por que não tentar uma IDE? Falei recentemente com Joel Sweatte na East Carolina University sobre o ensino de COBOL com a IDE do isCOBOL. A Veryant tem o prazer de fornecer a seus alunos licenças gratuitas para suas aulas. Ele diz que usar o ambiente baseado em Eclipse para seus programadores iniciantes não apenas torna o aprendizado mais fácil, mas eleva o COBOL ao mesmo nível que outras linguagens com as quais eles desejam trabalhar, como Java e C ++.

A Veryant está fazendo sua parte para resolver a falta de programadores COBOL, oferecendo licenças gratuitas para universidades

Não são apenas os alunos que preferem um ambiente de desenvolvimento gráfico. Estamos vendo um interesse cada vez maior na IDE do isCOBOL e, em seguida, movem-se para o mainframe para compilar e executar.

A Veryant está fazendo sua parte para resolver a falta de programadores COBOL, oferecendo licenças temporárias gratuitas para as universidades, como a East Carolina University e outras por meio da nossa conexão com COBOL Cowboys. Também estamos garantindo que mantemos nosso produto tecnologicamente avançado e fácil de usar, para que você possa manter seu COBOL, e ter suas aplicações modernas.



Russel Kirsch

O inventor do pixel de imagem digital morreu em 11 de agosto de 2020. Ele tinha 91 anos. O Sr. Kirsch usou uma foto digital de seu filho de 3 meses em 1957 para demonstrar como os computadores podem visualizar as fotos.

Secure web pages

Nossos sites são agora
<https://> páginas seguras, incluindo

www.veryant.com

e

support.veryant.com



CONTRATANDO

Se você estiver procurando por programadores COBOL, tente a página da comunidade da IBM “[Chamando todos os programadores COBOL](#)”. Você encontrará uma mistura de programadores de mainframe e não mainframe listados com credenciais e informações de contato.

YouTube Scription Drive
Inscreva-se em nosso
 [Canal do YouTube](#)

Dizendo tchau ao iscobol.debug. code_prefix

```
isCOBOL Shell 2020R1 (64 bit)

C:\Veryant\isCOBOL2020R1\sample\HELLO>iscc -d HELLO-WORLD.cbl

C:\Veryant\isCOBOL2020R1\sample\HELLO>dir
Volume in drive C is OS
Volume Serial Number is C0E1-8036

Directory of C:\Veryant\isCOBOL2020R1\sample\HELLO

10/20/2020 11:08 AM <DIR>      .
10/20/2020 11:08 AM <DIR>      ..
08/17/2020 02:32 PM          252 HELLO-WORLD.cbl
10/20/2020 11:08 AM      1,874 HELLO_WORLD$Debug$Infos.class
10/20/2020 11:08 AM      7,702 HELLO_WORLD.class
                                9,828 bytes
                                3 File(s)
                                2 Dir(s)  428,993,212,416 bytes free

C:\Veryant\isCOBOL2020R1\sample\HELLO>
```

Na 2020 R1 e anteriores, a depuração de programas isCOBOL só poderia ser feita se uma cópia do código-fonte fosse disponibilizada para a estrutura de tempo de execução. As variáveis de configuração seriam definidas para apontar para esses arquivos de origem.

`iscobol.debug.code_prefix` (to find the files locally)
`iscobol.debug.remote_source` (when debugging remotely)

Compilar com `-d` também resultou em uma classe adicional, no formato `<class name>$Debug$Infos.class`.

Tornamos a depuração muito mais fácil na 2020 R2, removendo a necessidade de configuração extra e arquivos de classe. Agora, quando você compila com `-d` ou `-dx`, o código-fonte é incluído no arquivo de classe compilado. Aqui estão algumas dicas para se lembrar quando usar este novo recurso de depuração:

- Para ver se uma classe foi compilada para depuração, execute o comando “`isrun -info <class name>`”
- A classe compilada para depuração será maior e executará mais lentamente do que uma não compilada para depuração.
- O código-fonte da classe é criptografado
- Você pode usar os mesmos comandos do depurador (“`br 7 HELLO-WORLD`” ainda define um ponto de interrupção na linha 7 do programa HELLO-WORLD).
- Você ainda pode usar as variáveis de configuração para carregar a fonte do disco para compatibilidade com versões anteriores

```
isCOBOL Shell 2020R2 (64 bit)

C:\Veryant\isCOBOL2020R2\sample\HELLO>iscc -d HELLO-WORLD.cbl

C:\Veryant\isCOBOL2020R2\sample\HELLO>dir
Volume in drive C is OS
Volume Serial Number is C0E1-8036

Directory of C:\Veryant\isCOBOL2020R2\sample\HELLO

10/20/2020 11:10 AM <DIR>      .
10/20/2020 11:10 AM <DIR>      ..
08/17/2020 02:32 PM          252 HELLO-WORLD.cbl
10/20/2020 11:10 AM      6,540 HELLO_WORLD.class
                                6,792 bytes
                                2 File(s)
                                2 Dir(s)  428,993,167,360 bytes free

C:\Veryant\isCOBOL2020R2\sample\HELLO>
```




Não usa a IDE do isCOBOL? É fácil começar.

Você ainda usa o prompt de comando para compilar seus programas? Bloco de notas ou V para codificar? Acho que você ficaria surpreso com a facilidade de mudar para a IDE do isCOBOL e como seus recursos são úteis. Aqui estão alguns recursos da IDE que acho que você vai adorar:

- O recurso de verificação de sintaxe em tempo real do IDE permite que você veja imediatamente os erros no código-fonte sem ter que compilar - Ao compilar, você saberá que não há erros de sintaxe
- Use o recurso Build Project para certificar-se de que compilou tudo. Por exemplo, se você tiver um projeto com 1.000 programas e alterar um arquivo de cópia usado por 50 desses programas (alguns com arquivos de cópia aninhados), o Build Project recompilará apenas os programas que usam o arquivo de cópia. Quando a compilação estiver concluída, é fácil atualizar seu ambiente de produção ordenando a pasta de saída por data e movendo apenas aqueles 50 programas.
- Você pode criar facilmente um processo para recompilar automaticamente todos os seus programas durante a noite - por exemplo, para testes internos necessários para o departamento de teste. Um exemplo de linha de comando para isso seria:

```
isIDE -data  
workspaceLocation -nosplash  
--launcher.suppressErrors  
-application com.iscobol.  
plugins.screenpainter.
```

IscobolScreenPainter.
builderApplication [project
projectName] refresh clean
build logfile logFilename

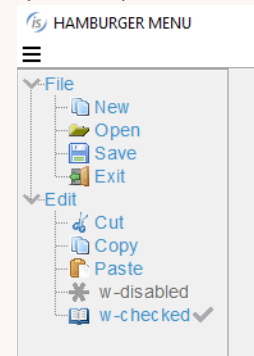
- A IDE se integra facilmente com software de controle de versão de código como o SVN, para que você possa manter suas alterações organizadas e seu código seguro.
- Você não precisa trabalhar fora do IDE para acessar suas atividades específicas do prompt de comando. As visualizações do terminal se conectam localmente ou por meio de SSH / Telnet de dentro do IDE
- Os hiperlinks podem mover você pelo seu código sem rolagem constante
- É útil visualizar seu código de maneiras diferentes - o IDE mostra uma visão geral, permite expandir ou recolher diferentes seções de código e dividir sua tela para visualizar duas partes diferentes de seu código ao mesmo tempo
- A IDE inclui uma pesquisa avançada e um campo de acesso rápido para que você possa encontrar qualquer coisa dentro ou fora do IDE
- Use o IDE no Mac e no Graphical Linux, bem como Windows. A IDE está incluso em seu sistema de desenvolvimento - sem nenhum custo extra envolvido. O suporte ao cliente terá o prazer de ajudá-lo a mudar para o IDE isCOBOL. Eu acho que, quando você se acostumar com isso, você vai se perguntar como você viveu sem ele!

Davide Spizzi,
Gerente de Engenharia de Suporte



NOVO HAMBURGER MENU

As telas de GUI costumavam ser consideradas um ambiente estático - você só precisava desenhar com um tamanho de tela em mente. Em 1999, [engenheiros da Microsoft disseram](#) que os computadores iriam “se tornar invisíveis”, ficando escondidos em objetos comuns, e isso certamente é verdade para muitos programas COBOL. É mais comum precisar escrever telas responsivas para que sejam exibidas em monitores, smartphones e tablets. Na Veryant, nosso objetivo é tornar isso o mais fácil possível para você.



A versão 2020R2 apresenta um menu de hambúrguer que pode ser implementado colocando uma variável de configuração em seu arquivo de propriedades em vez de alterar seu código.

COMO ALTERAR O ÍCONE DE SEU APLICATIVO

Você pode estar usando a sintaxe de ‘icon’ em sua instrução de janela de exibição para exibir um ícone na janela. Esse ícone também se torna o ícone da barra de tarefas.

Outra maneira de personalizar seu ícone é usar uma variável de configuração:

```
iscobol.gui.icon_file=<filename>
```

Você não precisa carregar ou descarregar o ícone com W \$ BITMAP, e atualizar seu logotipo é tão fácil quanto substituir o arquivo do ícone.



Evolution, without revolution

Veryant LLC

A equipe de suporte experiente e dedicada da Veryant está comprometida em fornecer os mais altos níveis de atendimento ao cliente

Para clientes com suporte, envie um e-mail para support@veryant.com

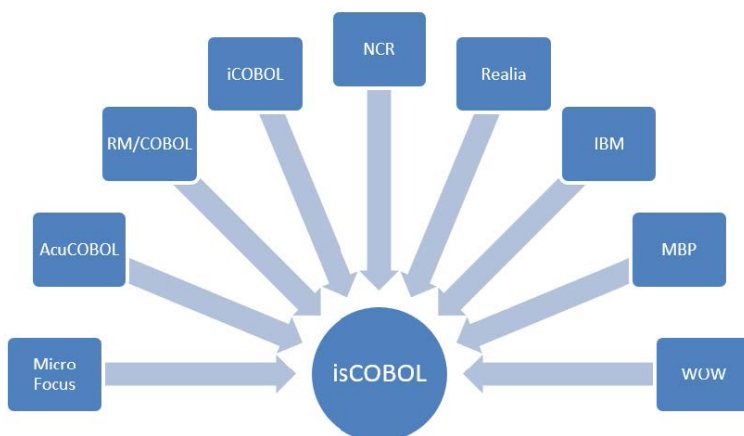
Se desejar que a Veryant entre em contato com você para agendar um briefing técnico do produto, envie um e-mail para info@veryant.com

Se desejar que a Veryant entre em contato com você para obter uma cotação especial ou assistentes de vendas, envie um e-mail para sales@veryant.com

Sede Corporativa
6390 Greenwich Dr., Suite 225
San Diego, CA 92122 - USA
Tel (English): +1 619 797 1323
Tel (Español): +1 619 453 0914

Sede Europeia
Via Pirandello, 29
29121 - Piacenza - Italy
Tel: +39 0523 490770
Fax: +39 0523 480784
emea@veryant.com

Como sempre, a 2020R2 contém vários acréscimos de compatibilidade - à medida que continuamos a tornar seu processo de conversão o mais simples, rápido e tranquilo possível.



veryant.com

Oferecendo escolha, flexibilidade e soluções econômicas para organizações com ativos COBOL valiosos

ENTRE EM CONTATO PARA UMA DEMONSTRAÇÃO

[**CLIQUE AQUI**](#)

Follow **Veryant** on



[**veryant.com**](http://veryant.com)