

Porque comentar código não é legal

18/08/2014 12:00

Opa, e aí moçada! Tudo certinho?

No post de hoje vou falar sobre um assunto que incomoda muita gente. E não é sobre elefantes. Na verdade, nunca entendi porque eles incomodam tanto. Bom, deixa pra lá e vamos ao que interessa.

Às vezes, quando analisamos programas antigos, podemos ver muitos trechos de códigos comentados. A ideia é substituir por novas linhas para melhorar o código ou alterar sua funcionalidade, como o exemplo abaixo:

```

9030  * 9436 - Re-determinação do tipo proc na tabela interna - 12.08.2014 - A
9031  * DEL.INI - 9436 GD - Atualização do tipo de operação
9032  *      READ TABLE ti_ynfe_cabec_xml INTO va_ynfe_cabec_xml WITH KEY check
9033  * DEL.FIM - 9436 GD - Atualização do tipo de operação
9034  * INS.INI - 9436 GD - Atualização do tipo de operação
9035  *      READ TABLE ti_ynfe_cabec_xml INTO lv_ynfe_cabec_xml WITH KEY check
9036  * INS.FIM - 9436 GD - Atualização do tipo de operação
9037  * IF sy-subrs EQ 0.
9038  *
9039  *      va_ynfe_cabec_xml-tp_proc          = lv_tp_proc.
9040  *      va_ynfe_cabec_xml-status_recfisico = lv_novo_status_recfisico.
9041  *      va_ynfe_cabec_xml-status_recfiscal = lv_novo_status_recfiscal.
9042  * DEL.FIM - 9436 GD - Atualização do tipo de operação
9043  * INS.INI - 9436 GD - Atualização do tipo de operação
9044  *      lv_ynfe_cabec_xml-tp_proc          = lv_tp_proc.
9045  *      lv_ynfe_cabec_xml-status_recfisico = lv_novo_status_recfisico.
9046  *      lv_ynfe_cabec_xml-status_recfiscal = lv_novo_status_recfiscal.
9047  * INS.FIM - 9436 GD - Atualização do tipo de operação
9048  * DEL.INI - 9436 GD - Atualização do tipo de operação
9049  *      MODIFY ti_ynfe_cabec_xml FROM va_ynfe_cabec_xml INDEX sy-tabix.
9050  * DEL.FIM - 9436 GD - Atualização do tipo de operação
9051  * INS.INI - 9436 GD - Atualização do tipo de operação
9052  *      MODIFY ti_ynfe_cabec_xml FROM lv_ynfe_cabec_xml INDEX sy-tabix.
9053  * INS.FIM - 9436 GD - Atualização do tipo de operação
9054  *
9055  * ENDF.
9056  * 9436 - Re-determinação do tipo proc na tabela interna - 12.08.2014 - A

```

Quem nunca teve que fazer isso? Se você perguntar o porquê, eis algumas possíveis respostas:

- Para melhorar a manutenção do código. Fazendo isso os outros programadores podem acompanhar as mudanças no código.
- É mais fácil de reverter para versões anteriores.
- Se você tiver seu cérebro devorado por algum zumbi, outro programador poderá acompanhar o que você estava fazendo.

Agora vamos deixar o bla bla bla de lado. Eu considero tudo isso um péssimo hábito. Primeiro, porque afeta a legibilidade do código, deixando mais complicado para fazer qualquer manutenção. Segundo, porque você simplesmente não precisa disso, é só usar o gerenciamento de versão.

O código fica cheio de comentários inúteis, o que sobrecarrega a leitura e análise do código. Como alguém disse [aqui](#) nesse fórum: “às vezes é como cavar um túmulo para encontrar o código que está sendo utilizado”. É, bem por aí. :/

O que fazer então?

Existe um gerenciamento de versão para a maioria dos objetos. Assim você pode comparar com a versão que está sendo utilizada em produção e também reverter para uma versão anterior ou ainda armazenar uma versão

intermediária. Então, o que eu sugiro é gerar uma versão antes de começar a alterar o programa. É só ir em Utilitários -> Versões -> Criar versão.

Comentar um código só faz sentido quando você não tem certeza se as alterações que você está fazendo estão corretas. Esses comentários podem te ajudar quando for testar o programa e entender porque as modificações não estão funcionando como esperado. Mas quando o programa for totalmente testado e está pronto para subir para produção, vale a pena gastar alguns minutos para deixar o código limpo. Também seria bom apagar variáveis ou rotinas que não estão mais sendo usadas.

“Então você acha que não deve ter nenhum comentário sobre as mudanças feitas no código?”

Não é bem assim, acho que um breve comentário no cabeçalho sobre as mudanças implementadas pode ajudar outros programadores a entender a funcionalidade atual do programa. Comentários úteis são sempre bem-vindos!

E qual sua opinião sobre o assunto? Comenta o código? Apaga tudo? Sai correndo? Deixa sua opinião ae! 😊
Valeu!

Comentários

Guilherme Dellagustin — 19/05/2015 18:49

Trabalho em um time com alguns devs mais velhos que carregam alguns hábitos de ABAPossauros, e um deles é o de comentar código e fazer o gerenciamento de versão na unha, sendo que temos controle de versão e diversos outros meios de documentar as alterações.

Eu sou veementemente contra, mas como é um processo do time, tenho que seguir também, o que é bem frustrante...

Pra mim código bom é código limpo e fácil de ler. Comentário serve pra documentar as exceções, aquela gabiarrarota e anti-intuitiva que você se viu obrigado a fazer (requerimento legal absurdo, otimização de performance, componente de reuso mal feito que você não tem controle), não pra controle de versão.

Billy — 02/11/2014 19:45

Concordo 100% com vcs... isso fode mais ainda a nossa vida de funcional, quando tentamos debugar...

Ótimo Post

Amauri Soares — 10/09/2014 16:24

Como ABAPer por apenas 6 meses, sempre perguntei o porque de tanto código comentado, se há controle de versão.

Agora entendi, posso ter o cérebro devorado por um zumbi...

Rafael — 08/09/2014 11:12

Concordo plenamente!

O que mais irrita então é quando vc pega um programa Z que é novo ainda (feito no projeto) e já ta cheio de

código comentado... P***, é um programa NOVO!!

E quando o código não tá indentado?! O dedo coça para apertar pretty printer haha

Mauro Laranjeira — 18/08/2014 14:51

MMaaaaaalaandro... Concordo 100%

Mas na grande maioria, os clientes exigem essa prática que foi implementada por desenvolvedores do cliente. Tipo dar um tiro no pé, pois estes desenvolvedores que darão a manutenção no código.

Muito bom post Dai

Abs,

Mauro Laranjeira

Alexandre — 18/08/2014 13:41

Se o comentário estiver no cabeçalho, tudo bem! O resto dos comentários atrapalham mesmo!

Ótima dica sobre as versões!