

Reescrevendo humanos em software: episódio #1

Tudo é software. Parece um mantra, mas é apenas uma verdade cada vez mais objetiva, em todo lugar que a gente vá, em toda coisa que se usa, em todas as funções econômicas e sociais. Não é algo aparente para uma vasta maioria das pessoas, ainda, mas é cada vez menos provável que alguém consiga passar um dia inteiro sem usar –ou estar inserido no contexto de– software, pelo menos em situações urbanas, nos países onde houver um mínimo de infraestrutura.

Exemplo? Imagine-se indo a um prédio de escritórios. Em quase todos eles, a portaria foi informatizada. Em muitas delas, o processo de recepção é mais ou menos o seguinte:

1. Vão capturar informação sua para um cadastro, incluindo uma foto;
2. Vão perguntar pra onde você vai e só vão lhe autorizar a ir até lá; como?
3. A catraca de acesso ao prédio lê seu crachá e indica, num visor, qual elevador você deve tomar;
4. O elevador não tem controles, apenas um display que indica onde está no momento;
5. O elevador lhe deixa no andar autorizado, como se tudo fosse mágica. Aliás, qualquer tecnologia suficientemente complexa [e que a gente não entende...] é indistinguível de mágica.

No parágrafo acima, você entrou num prédio de escritórios e uma representação sua “entrou” no sistema de informação da portaria. A informatização do prédio, na verdade, “cria” representações [abstratas] do prédio [concreto], cujo propósito é controlar, do mundo virtual onde estão os sistemas de informação, o que acontece no concreto, cá embaixo, onde estamos subindo e descendo no elevador.

Quer testar [o sistema de informação d]a segurança do prédio? Vá até o andar para o qual você está autorizado, ache a escada de incêndio e, por ela, se dirija a outro andar... para o qual, obviamente, você não tem licença para ir. **Se** [e um grande se, no Brasil, sempre ...] o **sistema** [agora, um grande sistema que inclui hardware, software e pessoas] funcionar, o alarme da porta deverá chamar a atenção de um humano para uma câmera de segurança, este acionará um guarda [armado, talvez] que sairá atrás de você, que fugiu dos limites impostos pelo **sistema**, criando uma exceção que deve ser tratada de pronto. Quanto mais rigorosas forem as regras do ambiente de que estamos falando, mas radical será a resposta do sistema como um todo, incluindo a dos agentes humanos dentro dele.

O blog não recomenda que você saia por aí testando o funcionamento [ou não] de sistemas como este, especialmente em ambientes onde você não detém o controle absoluto da situação. Pode ser perigoso e envolver riscos à vida.

Estamos usando **software** para construir sistemas que **simulam**, no **universo abstrato** da *computação, comunicação e controle*, o ambiente **concreto** de nossas ações do dia-a-dia, até nos casos simples como visitar um prédio. Tais simulações estão passando a condicionar, cada vez mais, a realidade, em quase todos os campos de atividade humana, determinando o que se pode ou não fazer em cada contexto, o tempo todo, em quase todo lugar.



Olhando pra imagem acima, a base de tudo são **linguagens** de programação, usadas para escrever um tipo peculiar de **texto**, chamado **programa**; programas são textos que representam algoritmos [sequências finitas de instruções bem definidas...] e que são **abstrações executáveis**: ao realizar as operações indicadas pelo programa, a “máquina” [um computador digital, quase sempre], “cria” **mundos virtuais**. Lá no começo do texto, o programa que informatiza a portaria virtualiza uma porção do prédio e do que nele acontece, fazendo com que, para um número de ações, só o que existe e pode acontecer no “mundo virtual” criado pelo software pode, de verdade, acontecer no prédio. Claro que há o *jeitinho brasileiro* em ação, em todo canto, mas isso é outra história.

O que nos interessa, pra fechar este episódio, é que **sistemas de informação** são **simulações de mundos...** e que estes **simulacros** estão, cada vez mais [em quantidade, intensidade, qualidade, grau de controle da situação...] presentes em todos os aspectos de nossas funções e interações pessoais, econômicas e sociais. Sim... mas o que é mesmo que isso tem a ver com reescrever pessoas em software?

Reescrevendo humanos em software:

episódio #2

No episódio anterior desta série, vimos que sistemas de informação, simulando o mundo ao qual estamos acostumados, começam a determinar o que podemos fazer de **concreto** em um ambiente controlado, cada vez mais, pelo **abstrato**. Além do mais, mostramos que “sistema”, na interpretação dada aqui, é mais que software, que é só parte da parada. No caso da portaria do prédio, o **sistema** tem catracas, crachás, elevadores, sensores [na escada de incêndio, por exemplo] e **humanos**: recepcionistas, supervisores, guardas, o **visitante**. Todo mundo que está “no sistema”, ou que interage com ele, faz parte do sistema de alguma forma.

Aliás, considerando que os humanos podem realizar *certas* sequências finitas de instruções bem definidas, é possível desenhar algoritmos que sejam computados **apenas** por humanos, sem auxílio de hardware, software e/ou outros artefatos. Antes da computação estar associada a computadores [as máquinas, hoje digitais], o termo **computador** era usado para descrever humanos que calculavam, quase sempre trabalhando em grupo para realizar cálculos complexos.

Hoje, *human computing* é o estudo do *desenho* e uso de esforço humano para realizar tarefas que computadores [**aqui, combinações de hardware e software**] ainda não podem realizar [e fazendo as quais os humanos, ainda por cima, podem se divertir]. Isso não tem quase nada a ver com *crowdsourcing*; enquanto esta substitui [no trabalho] trabalhadores comuns por humanos que fazem parte de alguma comunidade, aquela substitui ações que seriam realizadas por hardware e/ou software pelo esforço humano.

Para um número de coisas, *human computing* faz [muito] sentido, especialmente daqui pra frente. Algoritmos e sua implementação como software, e o “sistema” resultante, vão eliminar muitas posições de trabalho onde um humano, hoje, faz pouco mais que um trabalho repetitivo que poderia ser realizado por um robô. Quando [e aqui não há um “se”] reescrevermos estes trabalhos em software, o que será feito dos humanos que o realizavam? Um desenho de **computação humana** compatível com a natureza essencialmente humana de trabalhadores como nós pode ser uma saída de curto prazo *para quem perdeu o emprego para software*.

Mas... quem vai perder o emprego para software, e em que ordem?...

Primeiro, a tese que sustenta a argumentação, baseada em cinco marcos essenciais da história da informação em qualquer cenário onde haja vida inteligente.

Vamos lá: se há algum **1. Código da vida** [como DNA, aqui] em algum planeta, no universo, além do nosso, este “software” [sim, somos, no fundo, software], dentro de condições ambientais apropriadas [isto é, contexto “calmo” o suficiente para vida e aprendizado], será a base de um processo evolucionário que resultará em *organismos complexos* como os nossos. Os corpos destes seres terão **2. Cérebros** capazes de *entender e intervir* no ambiente. As primeiras intervenções, em ordem de complexidade, estarão relacionadas ao corpo, quase certamente codificando informação e conhecimento na forma de **3. Ferramentas**. Na terra, DNA existe há 3.5 bilhões de anos, e cérebros como os nossos surgiram há cerca de 4 milhões de anos; as ferramentas elementares para aumentar a capacidade de ferir [garras, ao invés de unhas] são datadas de 3.4 milhões de anos atrás, obra da família de Lucy, *oaustralopithecus afarensis*, em dikika, etiópia. As primeiras ferramentas, pelo que se acredita até aqui, são ossos pontiagudos, possivelmente modificados para tal aspecto, vistos na imagem abaixo.



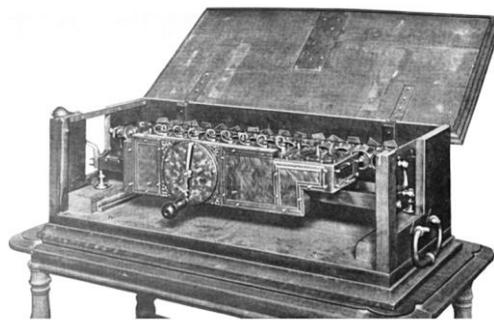
Ao certo, nunca saberemos exatamente onde e quando as primeiras ferramentas foram desenvolvidas ou “capturadas”. Pois parece que a capacidade de estender o próprio cérebro, registrando fatos relevantes da vida de forma externa ao corpo, na forma de **4. Texto**, leva muito mais tempo para acontecer. Da forma como usada hoje na maior parte do planeta, a escrita baseada em alfabetos como o nosso tem menos de 3.500 anos e vem da Grécia, ao redor de 1.400AC.

Comparado a seres rudimentares, sem acesso a ferramentas e linguagens escritas [sim, muitas: da poesia à ciência e tecnologia], **h. Sapiens** está no topo da cadeia do conhecimento. No séc. 17, começou a *descrever* o mundo de forma muito mais precisa do que até então e, ao mesmo tempo, a linguagem de descrição o tornou capaz de começar a *prescrever* mudanças antes inimagináveis. A sistematização da ciência e seus efeitos na técnica criaram, na prática, a indústria e sua revolução.

Textos para descrever e prescrever a fabricação e uso de ferramentas e sistemas [fábricas...] inteiros foram essenciais para codificar e disseminar conhecimento. A

industrialização do texto [Gutenberg] precede *watt* por mais de três séculos. O texto industrial, barato, commodity, possibilitou boa parte do que aconteceu nos séculos a partir dele. É difícil imaginar uma mudança da ordem da revolução industrial a partir de monges copistas... não é?

Não demorou muito até que, ainda no séc. 17, fosse possível começar a projetar e construir sistemas cujo propósito era substituir o cérebro, ainda mais quando o problema a ser resolvido envolvia muitas repetições de operações elementares. As calculadoras de Leibniz construídas entre 1674 e 1694, foram as primeiras máquinas a fazer as 4 operações sem auxílio humano ao calcular... mas era um humano que emprestava a força motriz para que o engenho funcionasse. Olhe a manivela, na imagem...



A máquina de Leibniz é primária, mas representativa de tecnologia como o domínio das possibilidades. Há 350 anos, Leibniz [e outros] já tentavam transferir o cálculo aritmético [que só humanos e talvez um cavalo realizam...] para as máquinas. Leibniz, Babbage e todos os que atacaram o problema do cálculo automático até a primeira metade do séc. 20 olhavam para o lugar certo... e viam as ferramentas e sistemas erradas. Era preciso criar uma nova classe de ferramenta, precedida por toda a teoria que a sustentasse.

E foi só na década de 30 que o princípio da máquina computacional universal, que consegue computar qualquer função que, em tese, pode ser computada, abriu caminho para a construção dos computadores “programáveis” e para a noção que temos, hoje, de programas e **5. Software**. A máquina “de Turing” [dos anos 1930] é a abstração que originou os computadores “de Von Neumann” [de 1945], cuja arquitetura ainda serve de base para quase todos os sistemas computacionais que usamos hoje.

Então, via **1. DNA, 2. Cérebros, 3. Ferramentas, 4. Texto e 5. Software**, chegamos onde queríamos. **Software** é o que especializa computadores universais para que seja possível, neles, computar tudo o que pode ser [eficientemente] computado. **programar computadores** – ou “fazer software”- é o que possibilita a criação de sistemas de informação que usamos para simular [ou reimplementar] o mundo. De acordo com a figura abaixo:



A partir deste ponto, você já pode começar a concluir o que só vamos escrever no próximo episódio: ao atingir o estágio software de sua sofisticação informacional, o que uma civilização vai fazer?... primeiro, software vai processar texto, porque, afinal, o próprio software é um tipo de texto; depois... bem, no próximo episódio a gente vai saber.

Reescrevendo pessoas em software: episódio #3

Conjectura: **toda** civilização evolui até o ponto em que começa a escrever **software**. Consequência: a partir daí, rapidamente, chegará ao ponto em que começará a escrever **pessoas** em **software**. Será que esta conjectura é uma verdade universal? Se for, como e por que acontece? E, do ponto de vista amplo, quais são os efeitos para as pessoas, a economia e a sociedade?

Tal discussão começou no **episódio #1** onde falamos de como certos tipos de texto, os programas, criam o software por trás dos sistemas de informação, as simulações de mundo que começam a “controlar” o mundo de verdade ao nosso redor. No episódio #2 contamos uma história da informação, que começa em DNA e, literalmente acaba em software. Como DNA pode ser considerado um tipo de código, também, a história da informação começa e termina em *software*.

dna » cérebros » ferramentas » texto » software

Já atingimos o ponto de chegada da conjectura. Talvez nunca saibamos se ela vale para todas as civilizações, pelo menos para as que evoluem ao ponto de escrever software antes que seu ambiente as inviabilize. Parece que não somos especiais, e o que fizemos e faremos deve ser comum aos que chegarem ao mesmo estágio. *A conjectura, assuma, é verdade universal.* Já que estamos falando de civilizações e que aqui só há uma [com variações, que devem se resolver nas próximas dezenas de milhares de anos], uma possível

consequência para encontros entre civilizações distantes [tipo anos-luz] seria que os visitados [em seu planeta] não o seriam por seres “vivos” de um mundo distante, mas por plataformas programáveis como o *curiosity rover*, o jipe que a NASA mandou para Marte. Logo, perca a esperança de ver alienígenas, de qualquer cor, no seu quintal, descendo de um disco voador. Porque, se nossa conjectura for verdadeira você vai se encontrar com software. Pelo menos pra começar. E [mas...] pode ser que o software “deles” já esteja aqui...

Agora, com o pé no chão: assim que começamos a escrever software, ele foi usado para processar texto; programas são um tipo de texto e escrever na linguagem “da máquina” não é econômico, para dizer o mínimo. Alguns dos primeiros programas tinham como função traduzir programas em linguagens mais parecidas com as que usamos [como inglês e português], que têm níveis de abstração superiores aos “de máquina”, para linguagem de comandos executáveis pelas máquinas. Pois o código executável pelos processadores não passa de sequências de zeros e uns. Desde as primeiras tabelas, planilhas, arquivos... até chegar em emeio e a web, com áudio e vídeo no pacote, software trata texto. Em 2011 foram 1.8 zettabyte tratados por software, sobre algum hardware [que foi usado como substrato programável].

Depois de texto, o próximo e óbvio passo era aplicar sistemas digitais ao controle de ferramentas e dispositivos que já eram regulados há tempos, como é o caso dos sistemas de realimentação gregos do **séc. 3 AC**. Máquinas, relógios e sistemas de todos os tipos começaram a depender cada vez mais de software na [e a partir da] segunda guerra e as tecnologias de sistemas embarcados, combinação de software e hardware utilizada para resolver problemas no mundo físico são quase ubíquas hoje. Sensores capturam sinais, software calcula o que deve ser feito em reação ou estímulo ao que está acontecendo e atuadores causam efeitos no ambiente. O ABS do seu carro é um tal sistema, o *air bag* idem. E todos os “controles” de todos os seus eletrodomésticos, da máquina de lavar roupas ao ar condicionado, também. No seu prédio, da bomba d’água ao elevador, passando pela porta da garagem, tudo é software. Tente viver sem ele...

Alan Turing, criador do conceito de máquina universal, programável, dizia que...*a ideia por trás dos computadores digitais pode ser explicada dizendo que estas máquinas têm o propósito de realizar qualquer operação que possa ser feita por um computador humano*. Desde o princípio, pois, na década de 50, se defendia a tese de que computadores [hardware+software] imitariam humanos no que nos é mais precioso, funções cerebrais associadas a, por exemplo, cálculo matemático. Pode não parecer tão “humano” como a Mona Lisa ou Brennan, mas, mesmo assim, é uma ideia poderosa, que criou todo um campo da ciência, a inteligência artificial. O debate [ou embate] ao redor do termo rola há décadas, mas há sinais de que resultados significativos começam a aparecer.

E demorou muito tempo pra se descobrir o que havia por trás da vida, de sua complexidade e diversidade. Watson e Crick chegaram à estrutura do DNA quase 20 anos depois de Turing estabelecer os princípios das máquinas universais. E só conseguimos sequenciar o genoma humano **50 anos depois**, em parte graças ao software usado para automatizar as ferramentas de sequenciamento e, depois, no processamento de texto [DNA é texto!]. O processo **não acabou...** e a engenharia reversa, completa, do código da vida, levará décadas. É por isso que ainda vamos levar algum tempo para **reescrever DNA** a sério, manipulando funções críticas dos corpos que quisermos modificar, como os nossos.

Reescrever o software dos humanos não vai rolar nos próximos poucos anos. E é o que dá a semântica do título da série: [ainda] não estamos prontos para **reescrever humanos**. Mas já estamos reescrevendo, agora, o *papel* que exercem na *economia e sociedade*. Estamos fazendo isso há tempos, processando texto e informatizando ferramentas. Como estes dois [e cérebros e DNA] codificam informação, nada mais natural do que usar software para tratar do que eles fazem. Daí, a sequência de uso de software na sociedade se escreve ao contrário do seu aparecimento...

software » texto » ferramentas » cérebros » dna

Como “reescreveremos humanos”?... do ponto de vista de seu papel na sociedade? Pode parecer complexo, mas não é. Veja o exemplo a seguir.

Agora que estamos quase no fim do ciclo de vida do carro urbano, talvez valha a pena lembrar dois momentos memoráveis da história do automóvel: Al Pacino, como Colonel Slade *em perfume de mulher*, pilotando [completamente cego] uma Ferrari Mondial *t em Nova Iorque com o auxílio de Charlie Simms*, vivido por Chris O'Donnell... a cena é inesquecível, especialmente a curva de 90 graus à esquerda.

Compare, 20 anos depois [Scent of a Woman é de 1992], o *Google Driverless Car* levando, sem piloto, Steve Mahan [cego, no lugar do motorista], pela cidade...:

No segundo vídeo, fica clara a tese: Google **reescreveu** o motorista. Em 20 anos, o *Colonel Slade* entrará numa Ferrari, sem Charlie, e irá onde quiser. Até pra Taperoá, onde o carro do Google Street View já foi duas vezes. Uma das razões, quero crer, é preparar o *Google Driverless Car* para venda em todo mundo. Afinal, como é que eu vou até Taperoá dormindo, de madrugada, se o sistema não conseguir chegar lá? Será que ele vai parar à beira da estrada e esperar que eu acorde? De jeito nenhum.

A comparação entre os dois filmes, seus tempos, a história contada nos episódios, até aqui, e nossa conjectura, diz o que vai acontecer: toda função humana para a qual 1. Humanos não tenham vocação inata e/ou 2. Haja perigo, para o humano que a

exerce ou outros, que estão sob sua influência, e/ou **3. Toda atividade** que não faça uso de funções mentais superiores para ser exercida será reescrita em software. *Assim que for possível. Ou seja, dentro das próximas décadas...*

Foi assim que digitador sumiu, como trabalho e profissão. É assim que o *motorista* vai desaparecer. Junto com *caixa de supermercado ou banco*. E mineiro de carvão, assim como *montador* de equipamentos eletrônicos. Viu a confusão que envolveu 2.000 empregados de uma fábrica da Foxconn na China? Nada menos de 5.000 policiais para controlar distúrbios numa fábrica de 79.000 empregados. Sabendo do trabalho robotizado e desumanizante que fazem, nenhuma surpresa, talvez só que não aconteça mais vezes.

Mas não é só o trabalho Chapliniano, que será transformado em software.

O astronauta e o neurocirurgião também serão reescritos. Os primeiros não fazem sentido. É mais econômico, seguro e simples... software explorar o cosmo, pois só precisamos sair daqui [e talvez, de vez...] quando soubermos para onde ir. No caso dos segundos, a precisão de qualquer um, por treinado e seguro que seja, é muito abaixo do que robôs podem fazer. É apenas questão de tempo para que muitos países comecem a *proibir neurocirurgias não mediadas por robôs*. Ou software. E robôs já reescrevem as práticas cirúrgicas, ao ampliar competências, reduzir riscos e tempo de recuperação. E custos [mas não, ainda, preços].

Um corolário imediato da conjectura deste texto é... *se você só realiza, no seu trabalho, atividades simples e que não exigem, agora ou no curto prazo, grande esforço cerebral, tranquilize-se: **você será reescrito em/por software***. É só uma questão de tempo. Tomara que seja antes da aposentadoria. Eu não confiaria nisso... e começaria a correr atrás de oportunidades para aprender mais, acima do que software pode e vai fazer no ambiente de trabalho, no tal do curto prazo.

Entre o presente e o futuro, claro que você pode escolher ser a favor ou contra o que vem por aí. Não se esqueça de ler Machado de Assis: segundo ele, o futuro nunca se engana. E nunca falha. E, com ele, vem **software**. Em todo lugar.

Vídeos citados:

Al Pacino: http://www.youtube.com/watch?feature=player_embedded&v=I3y2OvvuhDI

Self-Driving Car do Google:

http://www.youtube.com/watch?feature=player_embedded&v=cdgQpa1pUUE

Práticas Cirúrgicas com Robôs:

http://www.youtube.com/watch?feature=player_embedded&v=cdgQpa1pUUE