

Consistência de linguagens de interface

Consistência

A consistência é difícil de definir e portanto de medir, mas é intuitivamente reconhecida como um fator facilitador da aprendizagem. A vantagem da consistência reside no fato de **permitir a generalização** pelo usuário que, havendo aprendido parte do sistema, é capaz de inferir o resto.

Uma linguagem pode ser consistente em vários níveis diferentes, mas em cada nível as propriedades chave que determinam a consistência são **de configuração**, isto é, elas se relacionam com a estrutura toda da linguagem, e não com a natureza de relacionamentos tarefa-ação específicos.

Entendida a interface-usuário como linguagem, o argumento a favor da sua consistência reside em que ela permite que comandos e ações expressos pela interface possam ser descritos por meio de poucas regras, previsíveis e, portanto, fáceis de aprender e reter.

Exemplos de Linguagens

tarefa	linguagem 1	linguagem 2
mover cursor p/frente um caractere	CTRL-F	CTRL-L
mover cursor p/trás um caractere	CTRL-B	ESC-L
mover cursor p/frente uma palavra	ESC-F	CTRL-E
mover cursor p/trás uma palavra	ESC-B	ESC-E
ver próxima página	CTRL-V	CTRL-C
ver página anterior	ESC-V	ESC-C

A **linguagem 1** é parte da funcionalidade de um editor comercialmente disponível. Ela utiliza letras de forma mnemônica, mas tem problemas de consistência estrutural.

A **linguagem 2** não utiliza código mnemônico mas é organizada segundo um princípio único.

Níveis de consistência

Payne and Green (1986) descreveram, em suas gramáticas tarefa-ação (*Task-Action Grammars*), três níveis de consistência das linguagens de interface: semântico, sintático e lexical.

Enquanto os dois primeiros níveis (semântico e sintático) se referem à **consistência interna**, à expectativa determinada sobre uma parte da linguagem sobre o resto, a terceira (lexical) diz respeito à **consistência entre os itens lexicais e o mundo real** que eles representam para o usuário.

Aproveitando estes três níveis é possível estender esta proposta adicionando o nível discursivo, associado à consistência entre a estrutura que organiza e hierarquiza os elementos da linguagem e a expectativa do usuário relativa a esta organização.

1. Discursivo

O nível discursivo se refere à consistência estrutural, ou seja, à consistência entre a organização da funcionalidade da interface (ou da informação) e a forma como o usuário organiza mentalmente a mesma funcionalidade (ou informação).

É um tipo de **consistência entre a interface e o modelo mental que o usuário tem da tarefa** ou, dito de outra forma, entre a linguagem da interface e o modelo do usuário.

Quando os objetos e ações da interface guardam uma estrutura lógica que pode ser ancorada a objetos e tarefas familiares, a estrutura é guardada mais facilmente pelo usuário.

Uma interface é consistente do ponto de vista discursivo se sua estrutura funcional reflete a forma como o usuário organiza as tarefas correspondentes. Em outras palavras, uma interface é consistente neste nível se as suas funções estão organizadas de forma a refletir a forma como o usuário enxerga essas tarefas, implicando uma equivalência de níveis (por exemplo, o usuário tem disponíveis, ao iniciar uma sessão de certa aplicação, praticamente aquelas funções que ele esperava encontrar como grandes funções da ferramenta) e de granularidade (a subdivisão das funções no sistema correspondendo às diversas subtarefas enxergadas pelo usuário).

A consistência discursiva é conseguida por meio da participação ativa do usuário no processo de *design*, viabilizada pela adoção de técnicas de *design* participativo.

A consistência discursiva é pressuposto do uso bem-sucedido de uma interface para a consecução de uma tarefa que, de qualquer maneira, envolve um processo complexo.

2. Semântico

O nível semântico diz respeito à consistência **interna da linguagem de interface** relativa aos aspectos subjacentes à representação.

Ele engloba duas preocupações principais:

- a **completude semântica** da linguagem e
- a **consistência semântica** propriamente dita.

A **completude** é conseguida sempre que a linguagem tem poder expressivo completo, no sentido de permitir a expressão das nuances de significado por ela indicadas.

Uma linguagem de interface tem esse tipo de consistência semântica se ela atende à expectativa de expressividade por ela determinada.

Isto diz respeito a ações (por exemplo: “abrir” e “fechar”) da interface relacionadas do ponto de vista do significado, quando a disponibilização de uma leva à expectativa da disponibilidade das outras.

A **consistência propriamente dita** se traduz na ação consistente dos elementos da interface que efetivam a solicitação de tarefas/ações ao sistema.

Dito de outra forma, uma interface é semanticamente consistente se

- cada elemento da interface (botão, item de menu ou outro) atua de forma idêntica **em todas as situações de interação e sobre todos os objetos aos quais é aplicado** e
- cada tarefa/ação da interface é **efetivada a partir de uma única forma de expressão** (permitindo-se a solicitação de uma mesma ação por meio de elementos de natureza diferente, que devem guardar correspondência nos seus constituintes semelhantes)

3. Sintático

A consistência sintática é uma **consistência interna da linguagem**.

Nos tempos idos da Computação, os usuários necessitavam manter uma série de conhecimentos sintáticos de nível baixo, tais como as formas de apagar caracteres (<delete>, <backspace> e outras).

Estes detalhes sintáticos são dependentes de plataforma e, conseqüentemente, inconsistentes entre elas).

Um exemplo disso é o conjunto de comandos para o uso do correio eletrônico em um ambiente específico:

- <ENTER> para terminar um parágrafo,
- <CTRL> + <D> para encerrar a mensagem,
- Quit para sair do sistema.

O problema da dependência da plataforma torna-se maior pela coincidência parcial entre comandos, que leva à previsão, não-atendida, da consistência completa.

Embora as interfaces gráficas e o paradigma WIMP (*window, icons, menus and pointing devices*) tenham reduzido substancialmente a necessidade de o usuário lembrar detalhes de sintaxe, este conceito continua sendo crucial para os designers de interfaces.

O nível sintático associado às TAGs se referia às linguagens de comandos, estendidas para abarcarem as linguagens mistas compostas por teclas de função ou alfabéticas que davam conta das funções de edição do objeto do estudo.

O *design* de interfaces se aproxima mais facilmente da consistência se ele encara a interface de forma sistêmica, como linguagem e, assim, cria os seus componentes com preocupações “lingüísticas”.

Dentre os elementos da linguagem de interface que deve ter consistência sintática estão os seguintes:

- comandos tradicionais;
- rótulos (texto) de menus e botões;
- ícones.

Martins (1998) afirma que para que uma linguagem gráfica alcance um alto grau de comunicabilidade, ela necessita ser planejada de maneira sistêmica.

A percepção e a aprendizagem da sintaxe dos sistemas pelo usuário são difíceis pela arbitrariedade das escolhas anteriores, dependência de plataforma, pela dificuldade dos sistemas em proporcionar um modelo conceitual da funcionalidade como um todo, entre outros.

O desenvolvimento de um modelo conceitual que determine as diversas estruturas da linguagem facilita a percepção e a interpretação dos elementos da interface pelo usuário, seja esta textual e/ou gráfica.

Na concepção da linguagem nas interfaces gráficas (ou, mais precisamente, do tipo *WIMP*), é preciso levar em consideração alguns aspectos inerentes a este tipo de representação, entre os quais:

- Revelar a sintaxe da especificação de tarefas (exemplo: “objeto + ação” ou “ação + objeto”)
- Explicitar, na especificação da sintaxe a ser proporcionada ao usuário, operações como clique, clique duplo, arraste, soltura;
- Revelar a ordem de especificação de argumentos (exemplo: para apagar um objeto, deve-se arrastar o objeto até a lata de lixo ou a lata de lixo até o objeto?)

Este aspecto é muitas vezes resolvido “por tabela” pela semântica da tarefa, determinada pela metáfora utilizada (no caso de apagar jogar um objeto, em geral, leva-se o objeto até a lata de lixo, e não o contrário)

2.3. Alinhamento sintático-semântico

Os eixos semântico e sintático estão relacionados compondo um eixo intermediário, de consistência interna.

Uma linguagem tem alinhamento sintático-semântico se, além de ser consistente do ponto de vista sintático, ela guarda correpondência na ordem de especificação dos parâmetros, mantendo a consistência no eixo semântico.

Como exemplo, pode ser considerado o caso dos comandos de Cópia e Movimentação, ambos tendo o arquivo de origem (objeto) e o destino como parâmetros. Uma linguagem consistente no alinhamento sintático-semântico garantiria, não apenas a sintaxe geral dos comandos (comando + parâmetros) mais, também, a ordem de especificação destes, mantendo o par (objeto + destino).

Componentes do significado

A consistência relativa ao alinhamento sintático-semântico exige, também, que objetos e ações que envolvam várias dimensões de significado tenham os componentes comuns expressos da mesma forma, ou seja, que a estrutura sintática capture os aspectos semânticos comuns e revele os diferenciais.

Esta exigência deve se aplicar tanto a linguagens textuais quanto visuais.

Um exemplo desta consistência está na expressão gráfica de movimento, um segmento, com uma seta para a direita ou para a esquerda para indicar o sentido.

Linguagens de auxílio

Um caso particular de alinhamento sintático-semântico está no tratamento das mensagens de orientação das mensagens de erro como parte de uma única linguagem.

Neste intuito, para construir linguagens de auxílio (orientação e tratamento de erros), é necessário determinar:

1. quais os componentes semânticos da mensagem;

2. qual q estrutura em que eles serão apresentados ao usuário.

A adoção de uma linguagem simples e direta (quase teelgráfica) facilita a percepção e a interpretação das mensagens.

4. Lexical

A consistência neste nível se refere à relação **entre a representação do processo ou objeto na linguagem e a sua forma no mundo real**.

Esta forma de consistência é passível de aplicação tanto **nas linguagens de comandos** (valendo para os rótulos, que devem expressar de maneira precisa o significado da ação ou objeto subjacentes) quanto **nas linguagens gráficas** (que devem guardar uma relação não-arbitrária entre a representação e o significado subjacente).

Nas linguagens textuais (que englobam tanto as de comandos quanto aquelas subjacentes à rotulação dos itens de menu, de função, de botões, entre outras), a adoção da língua mãe do usuário endereçado pelo sistema produz, claramente, linguagens mais consistentes do ponto de vista lexical, já que uma palavra nessa língua têm um significado claro para este usuário, enquanto uma palavra em língua estrangeira guarda, na melhor das hipóteses (em que o usuário a domine), indireção na interpretação.

Nas interfaces gráficas, linguagens que utilizam ícones do tipo *Resemblance* são as mais consistentes, ao passo que a consistência vai se perdendo na direção dos ícones arbitrários, passando, no caminho, pelos ícones exemplares e simbólicos.

É uma falácia a afirmação de que representações gráficas são universais. Portanto, é preciso ter cuidado na adoção de representações, de forma a garantir que, na cultura do usuário potencial do sistema, estas sejam de fato interpretadas claramente conforma as expectativas do projetista.

Isto sem contar com os problemas das linguagens gráficas, que são extremamente dependentes de contexto, e esbarram na dificuldade de representar bem nuances de significado e aspectos dinâmicos, entre outros.

A adoção de padrões textuais ou gráficos e a exploração de relacionamentos de significado entre termos (como, por exemplo, em “abrir” e “fechar”) ou representações gráficas (“” e “”, claramente opostos), auxiliam à consistência neste eixo.

Decisões críticas envolvendo a consistência

As diretrizes são, como o nome indica, grandes linhas a servirem como norte no processo de design de interfaces. É sabido que elas não devem ser vistas como referências obrigatórias (até porque às vezes existe compromisso entre elas) nem utilizadas ao-pé-da-letra, mas, sim, devem ser interpretadas no domínio de tarefas e face ao usuário em questão. De qualquer forma, é claro que, via de regra, as diretrizes devem ser respeitadas a menos que haja algum motivo forte que justifique uma decisão em contrário.

Dentre todas as diretrizes, a consistência é, sem sombra de dúvidas, a mais visceral no processo de *design*, na medida em que a sua quebra pode comprometer a qualidade da interface, por melhor que sejam o estilo e o modelo utilizados.

Esta seção discute algumas situações nas quais a decisão da quebra da consistência, se não prescrita, pode pelo menos ser justificada.

a. Compromisso entre a consistência discursiva e a verificação diretrizes específicas

Exemplo: consistência discursiva (estrutura hierárquica de tarefas) X minimização dos passos na solicitação de ações freqüentes

Ações **Cortar**, **Copiar** e **Colar** na barra de ferramentas

b. Viabilização da inovação no artefato tecnológico

Exemplo: consistência discursiva (estrutura hierárquica) X acesso inovador às tarefas viabilizadas pelo sistema
acesso à resposta automática e ao encaminhamento de mensagens no correio eletrônico

O desejo de acomodar funcionalidade nova (inovações viabilizadas pelo artefato) no *design* pode entrar em compromisso com a consistência entre versões.

c. Interferência da linguagem do domínio de aplicação sobre a linguagem de interface

Exemplo: consistência no uso de rótulos em domínios onde um deles tem significado específico

ação Anular no sistema de urna eletrônica

d. Compromisso entre diferentes padrões (cultural, da plataforma, do tipo de aplicação, do domínio, entre outros)

Exemplo: O uso do recurso da cor em ambientes de *Computer-Aided-Design* na área de Geoprocessamento. Na cultura social, a cor verde é associada ao permitido, certo, correto. Esta associação é normalmente estendida aos ambientes computacionais. Mas este padrão se contrapõe, na Geografia Física, ao padrão do domínio, onde a cor verde indica floresta.

e. Compromisso entre a consistência em um nível específico e a adesão a um padrão

Exemplo: consistência na especificação sintática ou adoção de padrão de representação

Em um editor de texto, compor uma linguagem de comandos consistente para a ação de eliminação (dos objetos caractere, palavra, linha ou parágrafo), composta de:

Ação + objeto + sentido

Ou partir das teclas padrão para a eliminação de caracteres para trás <**Backspace**> e para a frente <**Delete**>, completando a funcionalidade de maneira semântica, sintática e lexicalmente inconsistente.

f. Quebra da consistência semântica para responder às necessidades de sistemas do tipo *life-critical* ou aos requisitos de segurança de acesso ou de uso

Exemplos: uso de recurso de *feedback* apenas no caso de ter que chamar a atenção para uma ação perigosa

não exibição (*echoing*) da senha (em oposição à exibição de qualquer entrada)

solicitação de confirmação unicamente para efetivação da ação Delete.