

3.6 ヒューマンエラー

エラーの性質

- 人は必ずエラーをする
- 絶対にエラーを許さないようなヒューマンインターフェイスは問題
 - エラーが検出できないようなシステム
 - エラーをするともう元に戻すことのできないシステム

Rasmussenの人間行為の3階層 モデルとエラー

- 技能ベースのレベルで起こるslip
- 規則ベースのレベルで起こるmistake
- 知識ベースのレベルで起こるmistake

技能ベーススリップ

- 行動が意図通りでないエラー
- 無意識に制御され、滑らかで自動的で、高度に統合されたふるまいにおいて起こる
- 比較的検出しやすく、スリップしたことにすぐ気が付くことが多い
- 気が散っていたり、注意が他にいつているときに起こる
- 技能ベースで行っている作業は多いので、スリップの発生数は多い
- スリップの発生確率はそれほど多くない

スリップのモード

- 二重捕獲スリップ
- 妨害に関する省略
- 意図の減少
- 干渉によるエラー
- 注意しすぎ

二重捕獲スリップ

因果関係はあるものの異なったものを捕獲

- 住所を書く際に、昔の住所を書いてしまう
- 大学に行く途中に買い物をしようと思ったが、気が付くと通り過ぎてしまった
- 寝室で普段着に着替えようとして、いつのまにかパジャマに着替えてしまった

妨害に関する省略

外部事象で混乱させられることによる失敗

- 本を買いに行ったら、本棚に面白そうな本を見つけた。しかし本来買うべき本を買い忘れた

意図の減少

意図が形成されたときと実施するときに時間遅れがあり、意図が減少

- 冷蔵庫を開けたが、何をしようとしたのか忘れてそのまま立っていた

干渉によるエラー

活性化している二つの計画などがあるために混乱

- ふたつのものを片付けようとして逆に置いてしまった

注意しすぎ

- 普段無意識にできるものが意識をしすぎるとどうやっていたのか忘れてしまう
- その結果、作業の省略、繰り返し、逆転などがおこる

ミステーク

- 正しい行為系列をつくることができないエラー
- 検出することが難しく、最後までミステークを起こしたことを気づかないことも多い
- 大きな危険を作り出す可能性がある
- 注意の焦点がはなれていなくてもミステークする

規則ベースミステーク

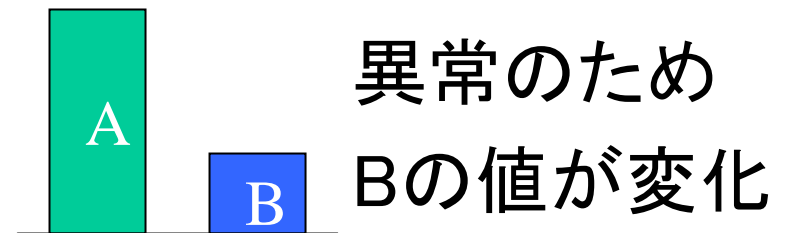
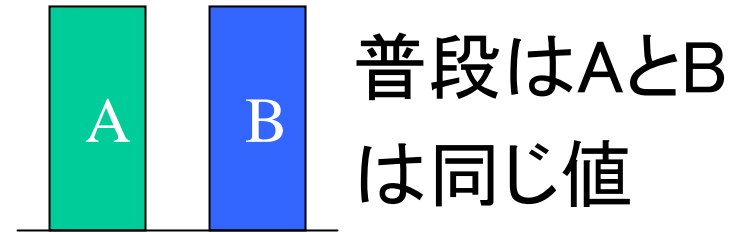
- 不適切な判断ルールを採用したために発生したミステーク
- 現在の状況に関するメンタルモデルを組み立てるときに、認知システムは存在する規則を組み合わせている。この規則も、カテゴリや構造化されている。ここで、誤った規則を組み立ててしまったり、規則は正しかったが適応方法を間違えたりしたときに起こる

正しい規則の適用の誤り

- ここでの「正しい規則」とは、特定の状況下での有効性が実証されているものである
- しかし、その状況と共通の特徴は持っているが、異なる一連の行動が必要となる要素を持っている環境条件では、誤って適用される危険がある

正しい規則の適用の誤りの例

原発の監視員が表示の値がAを示しているのにBを示すと勘違いした。これらの二つの値は普段は同じ値を示す。しかし異常が起こって、B値が変化しているのにAの示す数値が変化していないことから、警報が鳴り続けたが30分間そのミスに気が付かなかった



Aの値が変わっていないので異常ではないと勘違いをされた

誤った規則の適用

- 特に習得の途中などで、正しい規則が獲得できる前におきる
- 例 : 英語の過去形－edを使いはじめてから、不規則動詞にも－edをつける
- 例 : 運転が初心者の中には、速度制御とハンドル操作が同時にはできない

知識ベースミステーク

- 知識ベースで行動する際に、判断間違いなどで間違った行為系列を作ってしまう
- 問題解決を行う際に、持っている問題解決のルーチンを洗い出し、遅くて、逐次的で、労力が大きい意識処理によるオンライン作業となる
- この際にメンタルモデルを使用し、局所的な目標を達成するための行動を引き起こし、その行動がどの程度成功するかを観察し、目標との差を小さくするように修正しながら進んでいく
- 絶対数は少ないが、ミステークの出る頻度は高い
- 与えられた情報が不完全だったり、誤りを引き起こしやすい場合が多い

知識ベースミステーク

- エラーのモードとしても様々なものがある
- 正しい情報の取得ができなかった場合、判断を間違った場合など

エラー減少のための方策

人はエラーをすることを前提にして、ヒューマンインターフェイスを設計する必要がある

エラーの原因を理解し、その原因が最も少なくなるように設計する

- ミステークなら、与えられた情報が不完全だったり、誤りを引き起こしやすかったのだろう
- スリップであったなら、デザインが悪いか、ユーザが注意緩慢であったのだろう
- エラーの多くは論理的であり、理解可能である
- エラーをした人を責めたり、エラーを無視したりしてはならない

行為は元に戻すようにできること。
そうできないとしたら、元に戻せない
操作はやりにくくしておく

- 誤ってファイルを削除したりすることは頻繁に起こり、その損失は大きい。しかし、それを元に戻すようにできれば被害はほとんどない。
- やりにくい操作は誤って行うことも少なくなるので、危険の高い操作はやりにくくするとよい
 - 例えば削除するような操作は、簡単な操作ではできなくすることが望ましい

生じたエラーを発見しやすくしておくこと、また訂正しやすくしておくこと

- エラーに対して警告を発したりするなど、発見しやすくするとよい
- しかし、警告音だけでは多くの場合は解決策にはならない
 - ⇒ 無視されることが多い

- エラーに対する態度を変える。それを使っている人は作業を試みているのであって、そのために不完全ながら目標に近づいているのであると考えること。ユーザがエラーを犯していると考えべきではない
 - エラーの対策をするのもいいが、エラー対策のし過ぎのために使いにくくなってはいけない

強制選択法

エラーをしないようにする代表的方法の一つ
(物理的制約)

- 車のドアを外からキーで閉めなければロックできないようにすれば、鍵の車内への置き忘れは防げる
- 銀行や郵便局のCDやATMでは、お金を受け取る前にキャッシュカードを受け取らせることによって、カードの取り忘れを防止している

なんでも強制選択法を使えばよ いわけではない

- 強制選択法によって、正常な行動も影響を及ぼされることが多い
- 強制選択法が信頼性十分に動作しなければ、使いにくくなるだけ
- 強制選択をはずしたり無効にすることが頻発して役に立たなくなる

インターロック

- 操作が適切な順序で行われることを強制する、強制選択方法の一つ
- 安全分野で多く使用される
 - 消火器や手榴弾はピンをはずさなければ動作しないようになっている