

Action

(Cooper & Reinmann, 2003)

Flow and the Interface

- Flow (Csikszentmihalyi)
 - When people are able to concentrate wholeheartedly on an activity, they lose awareness of peripheral problems and distractions.
 - Software interaction should promote and enhance flow, rather than potentially breaking it and including flow-disturbing behavior.
 - The interface is the greatest threat!

Trouble with Interface

- An interface is
 - ...an artifact, not directly related to the goals of the user.
- The best interface is
 - ...no interface at all!
- Interfaces must be
 - ...at the service of the user, providing what is needed at every turn.

Making Interfaces Invisible

- Four guidelines:

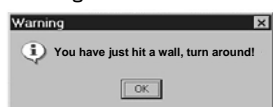
1. Follow mental models
2. Direct, don't discuss.
3. Keep tools close at hand.
4. Provide modeless feedback.

1. Follow mental models

- The mind looks for some **pattern of cause and effect** to gain insight into the machine's behavior.
- Provide what is **most likely the users will look for** first, based on where they are coming from.
- Yet, don't forget to improve on "mechanical-age" representations to **move things forward**.

2. Direct, don't discuss

- The ideal interaction is not a dialog – it's more **like using a tool** such as a hammer.
- **Direct feedback** is expected from the tool and the environment – not a dialog box.
- **Direct manipulation** idioms provide better and more flow-inducing interfaces.



3. Keep tools close at hand

- Most programs are **too complex for one mode** of direct manipulation to cover all features.
- **Tools offer different modes** of behavior for the program. (effectors, manipulators)
- Need to make tool manipulation and **switching easy** to prevent flow disturbance.
- Users should **not have to divert attention** from application to find a tool.

4. Modeless feedback

- Presented information and feedback should be **built into the main interface** and shouldn't stop the normal flow of activities.
- The **heads-up display (HUD)** is typically used for this purpose.



[Laurel, 1991]

Dramatic Techniques

- **Dramatic Theory**
 - Used to design interesting, engaging and satisfying human-computer activities.

Drama vs. Narrative

- **Human-Computer Activities**
 - More like plays than stories.

Stories	Plays
Description	Action (Enactment)
Detail	Intensity (Intensification)
Thematic Links	Causal Links (Unity of action)

Drama and Time



"...I'd design games that were meant to be played in **4-5 hours**. The games would be of the same scope that I currently design, I'd just **remove the silly time-wasting puzzles** and take the player for an intense ride. The experience they would leave with would be much more entertaining and a lot **less frustrating**." (Gilbert, "Monkey Island", 1989)

- **Drama imposes time limits**
 - So does an interactive system.

Constraints

- **Imposed constraints:**
 - "Real world" or hard-ware related.
 - "Mimetic world" or software related.

Creativity and Constraints

- **No limits**
 - Can produce a sense of powerlessness or even complete paralysis of imagination.
- **Limitations**
 - Paradoxically increase imaginative power by reducing open possibilities.

Creativity and Constraints

- **Closed Mimetic Worlds**
 - Provide a creative security net.
 - People respect this.
 - Increased potential for effective agency where causal relations are clear and not left open.

Good Constraints

- **How should constraints be expressed?**
- **Explicit**
 - Undisguised and directly available (e.g. menus)
 - Can be used before action.
- **Implicit**
 - Inferred from behavior of system (e.g. failing)
 - Preferred during the course of action.

Good Constraints

- **The Mimetic Context**

- Can provide a way to make constraints appear to be within the scope of the activity.



Tutorial in Star-Trek
Elite Forces

The Power of Context

- **Constraints should limit...**

- ...**not** what we can do,
- ...but what we are **likely to think of doing**.

- **Context**

- Is the most effective way to do this.

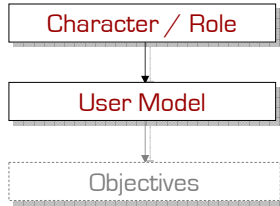
Character as Context

- **Character**

- A role template giving rise to action.
- Implicitly constrains choices.
- Inspires creativity in fulfilling that role.



User as Character



Plan Recognition

- **Discover**
 - What action is being performed (observation).
 - What process has started (inference).
 - What objectives are being pursued (user model).
- **Intervene**
 - To help user fulfil their role.
 - To guide user to a different role.

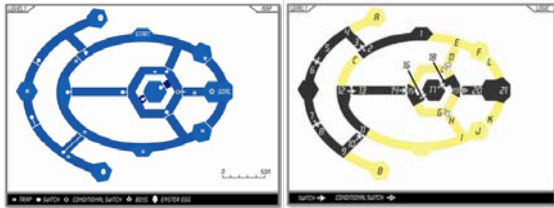
(Bergeron, 2006)

Action and Progression

- Progression and challenge need to be tied together.
- You receive the motivation, the knowledge and the skill to take on each level of challenge.
- Challenge can be broken up into levels, which can be described with level maps and logic maps.

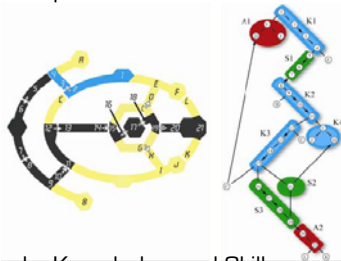
Action and Progression

- Level Maps



Action and Progression

- Logic maps and trees



- Attitude, Knowledge and Skill

References

- Cooper, A., Reimann, R., Reimann, R. M., and Dubberly, H. 2003 *About Face 2.0: the Essentials of Interaction Design*. John Wiley & Sons, Inc.
- Laurel, B. 1993 *Computers As Theatre*. 2nd. Addison-Wesley Longman Publishing Co., Inc.
- Bergeron, B. 2006 *Developing Serious Games*. Charles River Media, Inc.
