

# L'interaction est l'avenir de l'informatique

Michel Beaudouin-Lafon  
in|situ|  
15 novembre 2006  
mbl@lri.fr - insitu.lri.fr

## Réflexions sur 10 ans de l'AFIHM



Où en sommes-nous ?



Où allons-nous ?

Quelle est notre discipline ?

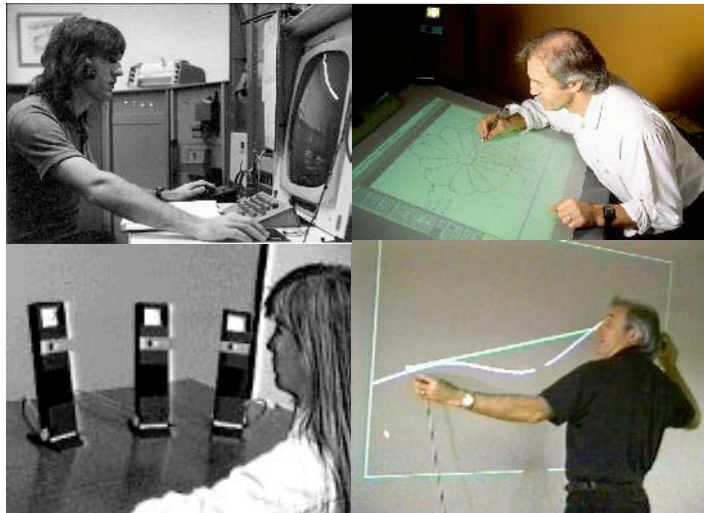
Deux idées dans cet exposé :

- Nous avons besoin de visions ... et de visionnaires
- L'interaction est un paradigme majeur de l'informatique

Doug Engelbart	
Augmenting human intellect	
	

NLS/Augment	
	


Bill Buxton  
Interaction Design



Digital tape drawing



Peter Wegner  
Interaction vs. Algorithms Comm. ACM, mai 1997



**Why Interaction Is More Powerful Than Algorithms**

Peter Wegner

Interaction is a more powerful paradigm than rule-based algorithms for computer problem solving, overturning the prevailing view that all computing is expressible as algorithms.

THE PARADIGM SHIFT FROM ALGORITHMS TO INTERACTION captures the technology shift from mainframes to workstations and networks, from number-crunching to embedded systems and graphical user interfaces, and from procedure-oriented to object-based and distributed programming. The radical notion that interactive systems are more powerful problem-solving engines than algorithms is the basis for a new paradigm for computing technology built around the unifying concept of interaction.

**From Sales Contracts to Marriage Contracts**

The evolution of computer technology from the 1970s to the 1990s is captured by a paradigm shift from algorithms to interaction. Algorithms yield outputs completely determined by their inputs, while interactive systems, like PCs, online services, and robots, provide history-dependent services over time that can learn from and adapt to experience.


Algorithms are "sales contracts" delivering an output in exchange for an input, while objects are ongoing "marriage contracts." An object's contract with its clients specifies its behavior for all contingencies of interaction (in sickness and in health) over the lifetime of the object (ill death do us part) [8].

**UIST '02**  
15th annual symposium  
User Interface Software & Technology

Paris, France  
27-30 oct. 2002  
[info@uist.org](mailto:info@uist.org)

**Home**  
Program  
Registration  
Venue  
Contest  
Committees  
Sponsorship  
Student Volunteers  
Call for participation  
Previous UIST

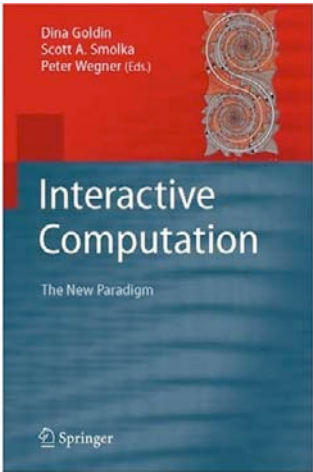
**Peter Wegner**  
Brown University



**Models of Interactive Computation**

The talk will open with a brief outline of aspects of my career, including work on programming languages, software engineering and object-oriented programming. I will present my approach to programming languages, software engineering technology, and object-oriented programming methods as tools for problem-solving.

Next, I will review Turing's work in the 1930s on Turing machines and the Church-Turing thesis, and its mistaken adoption in the 1960s as a universal model of computer problem-solving. This will be followed by a review of my recent papers dealing with interactive computation as a more expressive paradigm than that of Turing machines, better able to express problem-solving in software technology and user interfaces.

<h2>Interactive Computation / HCI Remixed</h2>	
	<p><b>Tom Erickson</b> · home · pubs · essays · HCI Remixed</p> <h3>HCI Remixed:</h3> <p><b>Page Contents</b></p> <p><a href="#">Public Call for Participation</a> <a href="#">If you're interested, please read</a></p> <ul style="list-style-type: none"><li>• <a href="#">About the HCI Remixed Project</a> (The project goals and FAQ)</li><li>• <a href="#">About the HCI Remixed Process</a> (The submission process and</li><li>• <a href="#">Shortcut to resources</a>, including the <i>Works Taken</i>' list)</li></ul> <p><b>Public Call for Participation</b></p> <p>We are inviting you to contribute to a unique project for the HCI community: researchers and practitioners reflect on a paper or other piece of work by someone on their view of or approach to HCI.</p> <p>We intend the essays to be brief (up to about 1800 words), personal in tone, and rather than taking a critical stance the goal is for each essay to examine the paper. The work will provide a unique perspective on how HCI has developed, will directly remind them that their own work, by influencing others, can have a lasting impact.</p> <p>The status of the project is that we've signed a letter of agreement with MIT Press and also intend to launch a blog associated with the project. A full description of the project (FAQ), submission process, and deadlines follows.</p>

<h2>Le paradigme de l'interaction selon Wegner</h2>
<p>Un modèle de calcul fondé sur des systèmes communicants</p> <p>Chaque système fournit des services</p> <ul style="list-style-type: none"><li>- en prenant des entrées issues d'autres systèmes</li><li>- et produisant des sorties vers d'autres systèmes</li></ul> <p>Tous les systèmes ne sont pas informatiques</p> <p>On s'intéresse à un système, pas à l'ensemble</p>

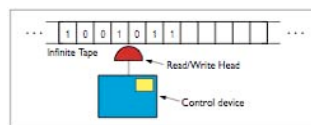
## Le paradigme de l'interaction selon Wegner

<i>Interaction</i>	<i>Algorithme</i>
Résolution de problème	Calcul algorithmique
Comportement observable	Transformation "boîte noire"
Système ouvert	Système fermé
Concurrence	Séquence
Contrat de mariage	Contrat de vente

## Une idée radicale

### Machines interactives

plus puissantes que n'importe quelle machine de Turing  
(en tirant parti de la puissance de l'environnement)



### Remise en question de la thèse de Church

un ordinateur peut faire plus que "les fonctions calculables"

## Une position philosophique

### Rationalisme / Positivisme

L'ordinateur est une machine déterministe

On peut parfaitement contrôler son fonctionnement  
(mais alors pourquoi y a-t-il toujours des bugs ?)

### Empirisme

Sacrifie la complétude et la prédictibilité

pour traiter des problèmes plus intéressants (plus réels)

Créer des systèmes ouverts

=> la fin du *mythe de la perfection*

## Une évidence dans l'informatique d'aujourd'hui

Réseaux de communication

Grilles de calcul

Réseaux d'agents

Algorithmes probabilistes, algorithmes anytime

Systèmes complexes

... et les systèmes interactifs homme-machine (SIHM) !

## SIHM : résolution de problème

Limites de l'analyse de tâches :  
l'humain ne se conforme pas à des modèles d'activité

Ordinateur-outil

Interfaces pour la création  
et la créativité

Appropriation

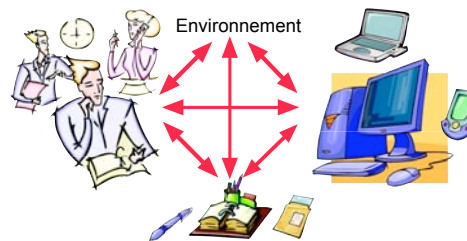
Social computing  
exploiter les capacités de la collectivité (ex: Wikipedia)



## SIHM : des flux dynamiques

Dépendances

- entre sorties présentes et entrées futures  
actions de l'utilisateur
- entre entrées présentes et sorties futures  
feed-back, feed-forward, ...
- entre le système et son environnement  
cohérence entre état interne et externe



Observabilité de l'état de l'interface  
Événements à de multiples niveaux d'abstraction



## SIHM : rôle de l'environnement

Un SIHM sans utilisateur n'a aucun intérêt !

L'environnement joue un rôle croissant :

- réalité mixte et augmentée
- interfaces tangibles
- ubicomp / pervasive / ambient computing

L'environnement est non calculable

Action située (Suchman)

Co-adaptation (Mackay)



## SIHM : concurrence

Parallélisme entre actions du système et de l'utilisateur

Aspect critique de l'utilisabilité : boucle action-perception, indicateurs de progression, animations, etc.

Interfaces réparties

Interaction multi-surfaces



## Interaction : quelques défis

Manque de modèles d'interaction (h-m)  
pour raisonner à un plus haut niveau d'abstraction



Manque d'outils de programmation  
adaptés à l'interaction

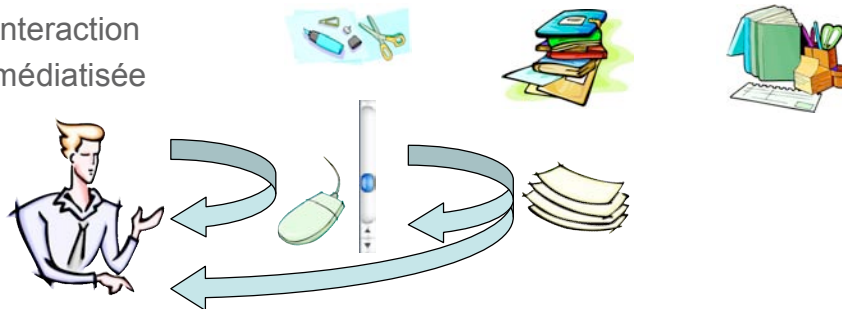


Manque d'outils d'évaluation  
adaptés à l'approche empirique



## Exemple de modèle : l'interaction instrumentale

Interaction  
médiatisée



Principes de conception :

réification, polymorphisme, réutilisation

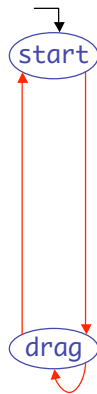
avec W. Mackay

=> *Approche générative de la conception*

ex : palette bimanuelle, guides magnétiques

avec C. Appert

## Exemple d'outil de programmation : SwingStates

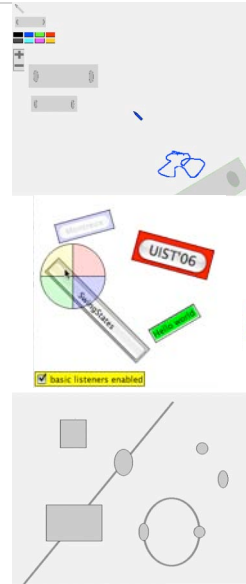


```

    StateMachine sm = new StateMachine("machine") {
        SShape dragged;

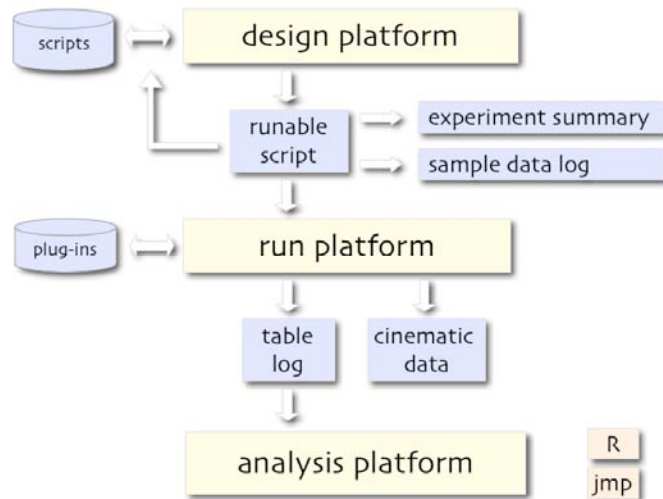
        public State start = new State() {
            Transition on =
                new PressOnShape(">> drag") {
                    public void action() {
                        dragged = getShape(); }
                };
        };

        public State drag = new State() {
            public void enter() {
                dragged.highlight();
            }
            Transition in = new Drag() {
                public void action() {
                    dragged.move()
                }
            };
            Transition off =
                new Release(">> start") { };
            public void leave() {
                dragged.unhighlight()
            }
        };
    };
    
```



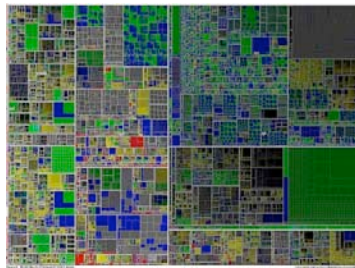
avec W. Mackay, C. Appert,  
 JD. Fekete, O. Chapuis, Y. Guiard

## Exemple d'outil d'évaluation : TouchStone



## Conclusion

Un changement de paradigme en informatique :  
de l'algorithme à l'interaction !



Soyez-en les acteurs !

Merci  
Questions ?

