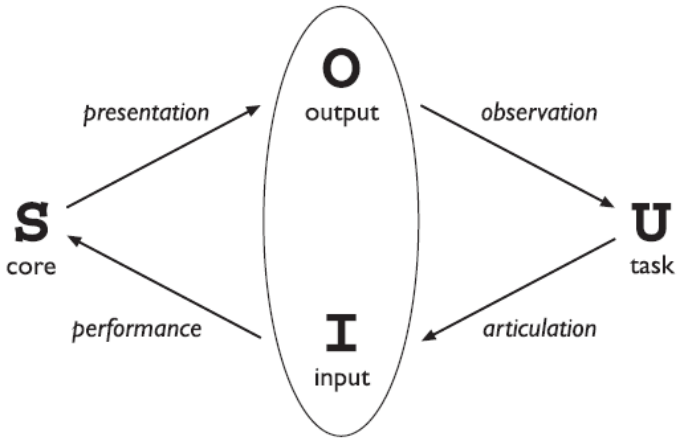
Abowd and Beale Interaction Framework

- Norman's model is a useful means of understanding the interaction, in a way that is clear and intuitive. It allows other, more detailed, empirical and analytic work to be placed within a common framework. However, it only considers the system as far as the interface. It concentrates wholly on the user's view of the interaction. It does not attempt to deal with the system's communication through the interface. An extension of Norman's model, proposed by Abowd and Beale, addresses this problem.

- The interaction framework attempts a more realistic description of interaction by including the system explicitly, and breaks it into four main components, as shown in Figure below.



The interaction framework identifies four steps in the interaction cycle:

1. the user formulates the goal and a task to achieve that goal
2. the interface translates the input language into the core language
3. the system transforms itself into a new state
4. the system renders the new state in the output language and sends it to the user

Translations
The four translations involved in the interaction framework are

1. articulation - the user articulates the task in the input language
2. performance - the interface translates the input language into stimuli for the system
3. presentation - the system presents the results in the output language
4. observation - the user translates the output language into personal understanding

Articulation
Articulation is the user's translation of their task into the input language. Consider a user who wants to turn on the lights at the far end of a room.

Poor Articulation
The user sees a bank of unlabeled switches. The user has no idea which switch controls the lights at the far end of the room. The problem here is that the language provided (the unlabeled switches) does not allow the user to articulate the goal (turn on the lights at the far end of the room).

Good Articulation
The switches are clearly labeled. The user articulates their task of pressing the switch that is labeled "the far end of the room". The language provided here (the labeled switches) allows the user to articulate their task without difficulty.

Performance
Performance is the interface's translation of the input language into stimuli to the system. This translation is determined by the designer or programmer (not the user).

Poor Performance
Consider a remote control for a television without a button for turning off the television. The user must go directly to the device and turn it off on the control panel.

Presentation
Presentation is the translation of the system's new state into the output language of the interface. This translation is determined by the designer or programmer.

Poor Presentation
Consider writing an essay using a word processor. You need to see the effects of your editing as a whole. However, the word processor only displays the immediate paragraph without the surrounding text or other pages. The surrounding text and other pages may have changed as well during the editing. In effect, all of the state changes cannot be displayed in the output language. You cannot readily confirm the effect of your editing throughout the essay.

Observation
Observation is the translation of the output language into personal understanding. This translation is done by the user.

EXAMPLE
Consider some deficiencies that might arise in configuring a game:

- Articulatory Problem - the user is not sure which options to set in order to configure the game properly
- Performance Problem - the controller does not have the ability to select the option
- Presentation Problem - the display does not show that the option has been set
- Observational Problem - the user does not interpret the display correctly

Anyone of these deficiencies would give rise to an interaction problem.