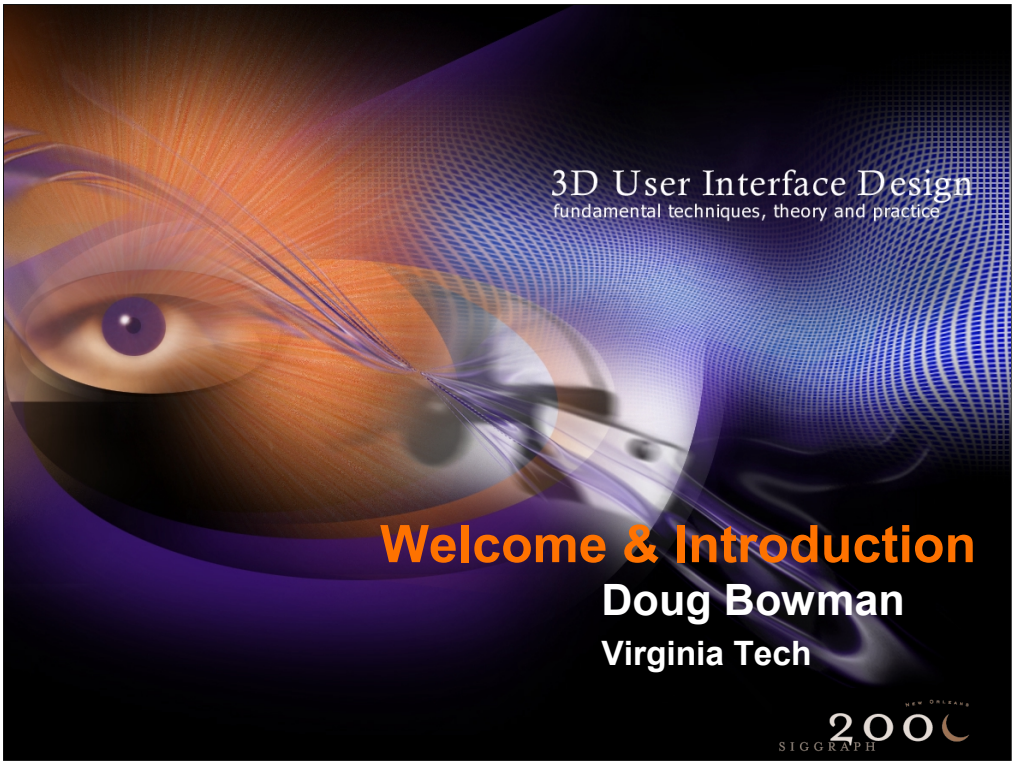
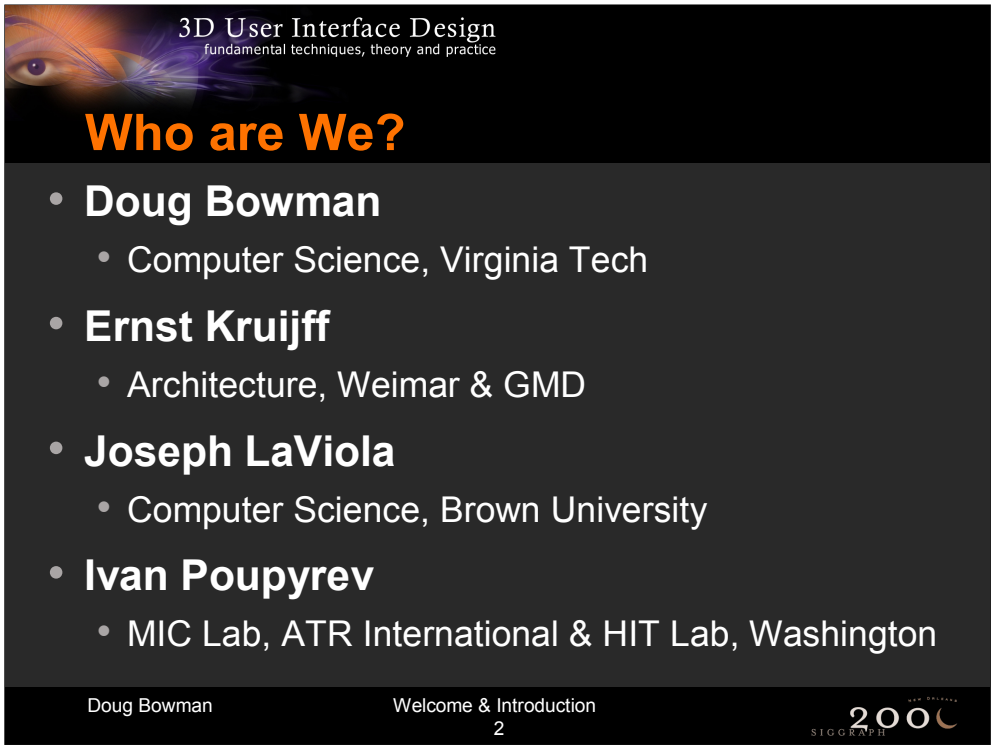


Welcome to the 3D User Interface (3DUI) course at SIGGRAPH 2000. The organizer of this course is Doug Bowman of Virginia Tech. In this section, you will meet the other presenters and get an overview of the topics we will cover during the course.

My contact information:

Doug Bowman
Dept. of Computer Science (0106)
660 McBryde Hall
Virginia Tech
Blacksburg, VA 24061
bowman@vt.edu
http://www.cs.vt.edu/~bowman/

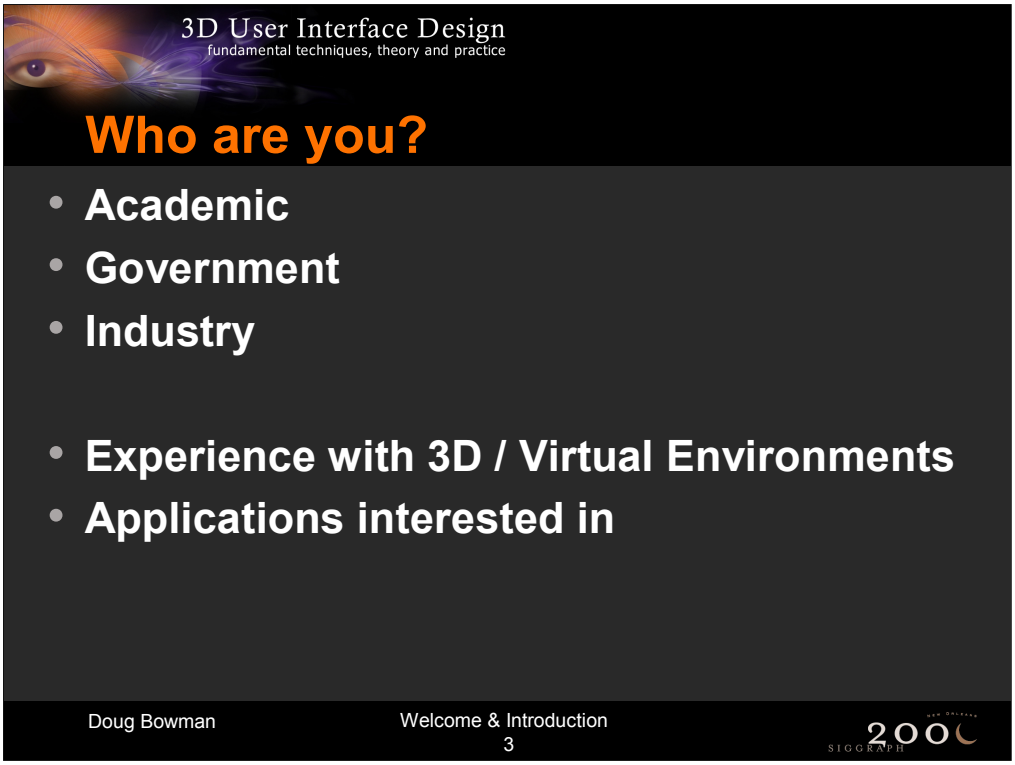


Doug Bowman
Assistant Professor of Computer Science
Virginia Polytechnic Institute & State University
bowman@vt.edu

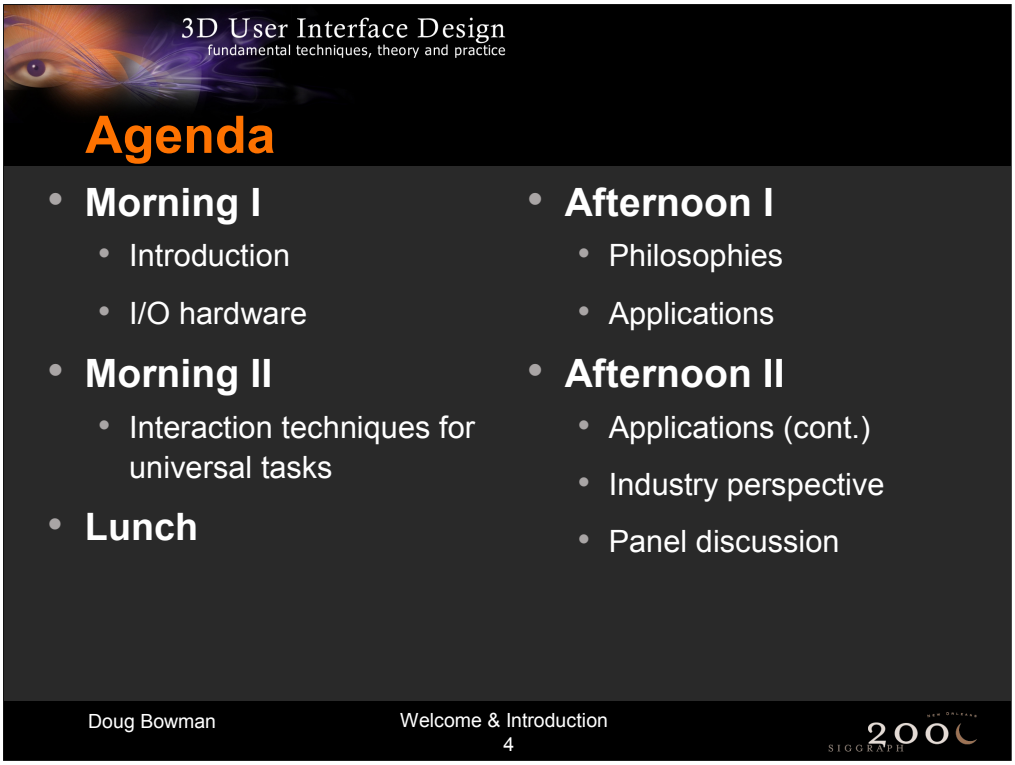
Ernst Kruijff
Bauhaus University - Weimar
Working with Prof. Donath
Also guest researcher at GMD
kruijff@archit.uni-weimar.de

Joseph LaViola
Ph.D. student, Brown University
Working with Dr. Andries van Dam
jjl@cs.brown.edu

Ivan Poupyrev
Researcher at ATR International, MIC labs
formerly at Human Interface Technology lab, Univ. of Washington
poup@mic.atr.co.jp

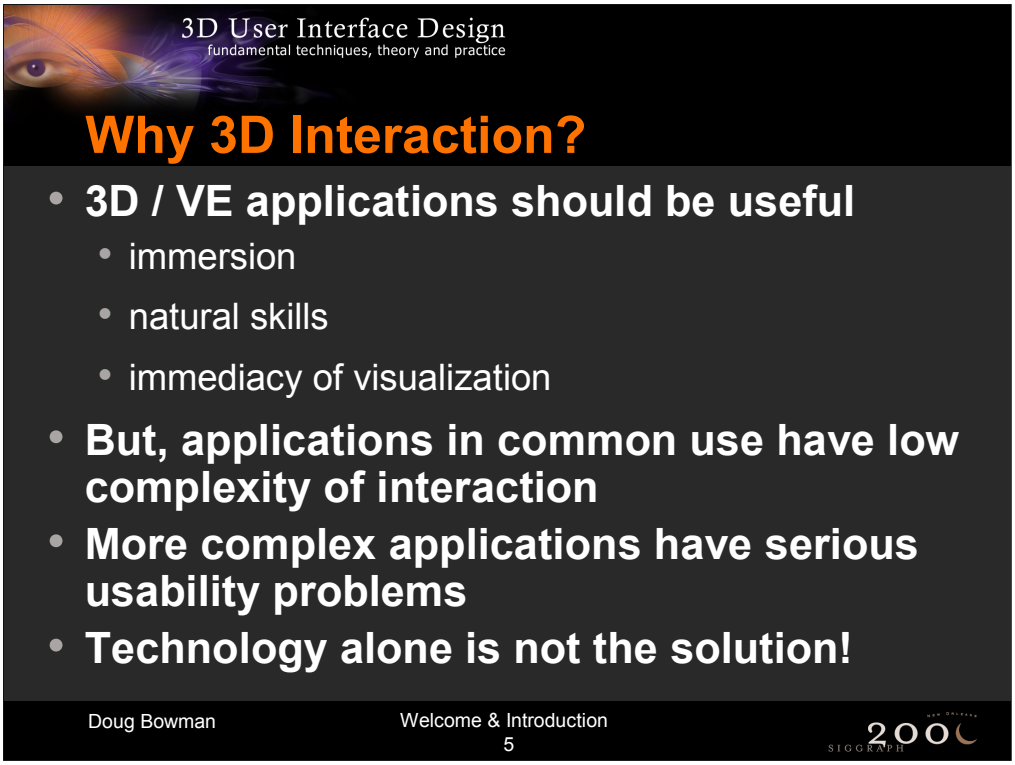


We hope to have a mix of backgrounds, levels of experience, and application areas in the course. We will attempt to keep a focus on applications during the course so that it is easy to apply the information and techniques we present.



Schedule:

8:30-10:00	Morning I
10:00-10:15	Break
10:15-12:00	Morning II
12:00-1:30	Lunch
1:30-3:00	Afternoon I
3:00-3:15	Break
3:15-5:00	Afternoon II



Three dimensions and virtual environments intuitively make sense for a wide range of applications, because of the characteristics of the tasks and their match with the characteristics of these environments. Immersion is the feeling of “being there” (replacing the physical environment with the virtual one), which makes sense for applications such as training and simulation. If a user is immersed *and* can interact using natural skills, then the application can take advantage of the fact that the user already has a great deal of knowledge about the world. The immediacy characteristic refers to the fact that there is a short “distance” between a user’s action and the system’s feedback that shows the result of that action. This can allow users to build up complex mental models of how a simulation works, for example.

Some applications in common use, each of which contain user interaction which is not very complex:

-design verifications, such as architectural walkthrough, and other passive visualizations

-phobia treatment (also very passive for the user)

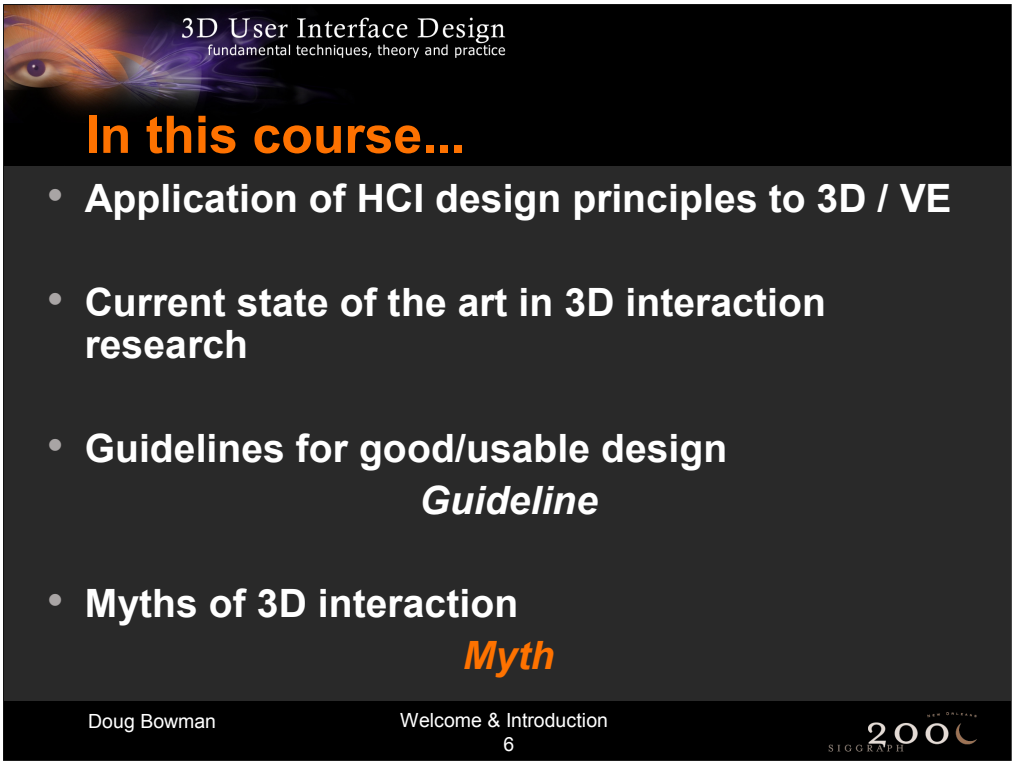
-entertainment (e.g. point and shoot at enemies)

-training (usually only navigation is required)

Other types of applications (e.g. immersive design, education, complex scientific visualizations) are for the most part still stuck in the research lab, often because they have usability problems that limit their usefulness.

Better technology is not the only answer - 30 years of VE technology research have not ensured that today’s VEs are usable - we must also focus on the design of interaction for VEs.

Therefore, we feel that 3D interaction is a vital topic for all 3D/VE developers, designers, and evaluators to understand.

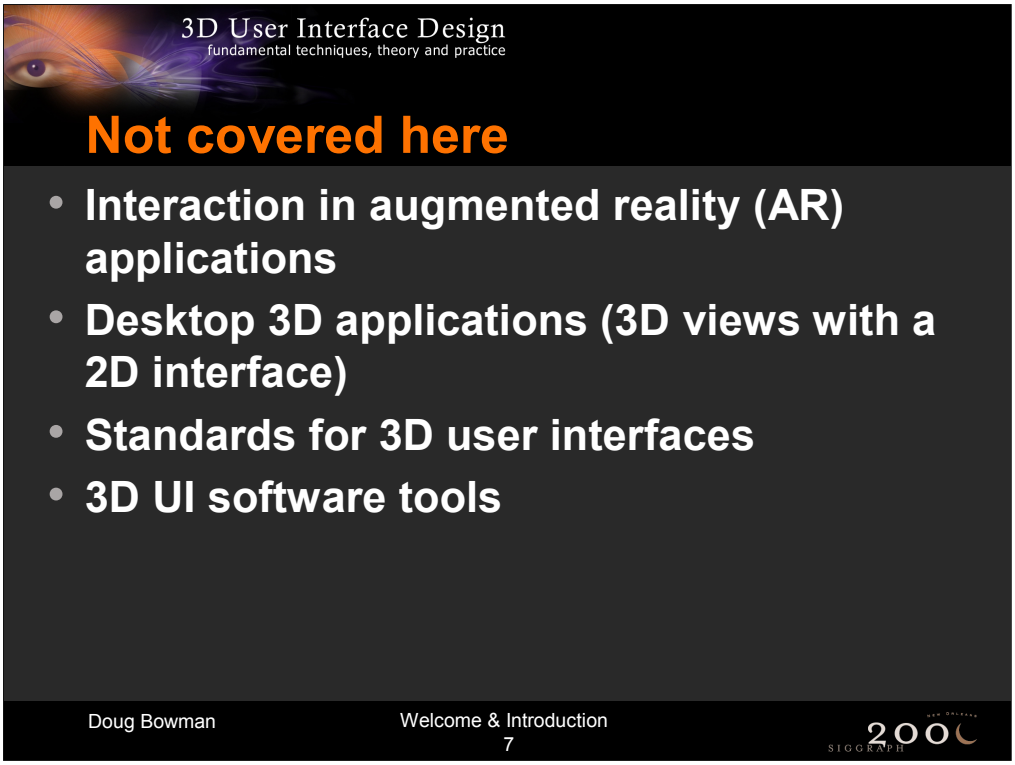


This course will not simply dump a lot of unstructured information on you. Rather, we have several broad goals in this course.

First, we will always be looking back, in a sense. There exists a large amount of knowledge about the design of 2D interfaces in the field of human-computer interaction (HCI). While these guidelines may not specifically apply to 3D interfaces, the general principles behind the guidelines still hold true. Therefore, we attempt to generalize and apply HCI design ideas to the area of 3D and VE user interfaces.

Second, we will present a large amount of information about the current state of the art in 3D interaction research. We structure this information by first distinguishing between hardware and software technologies, and second talking about some universal tasks for which we need interaction techniques.

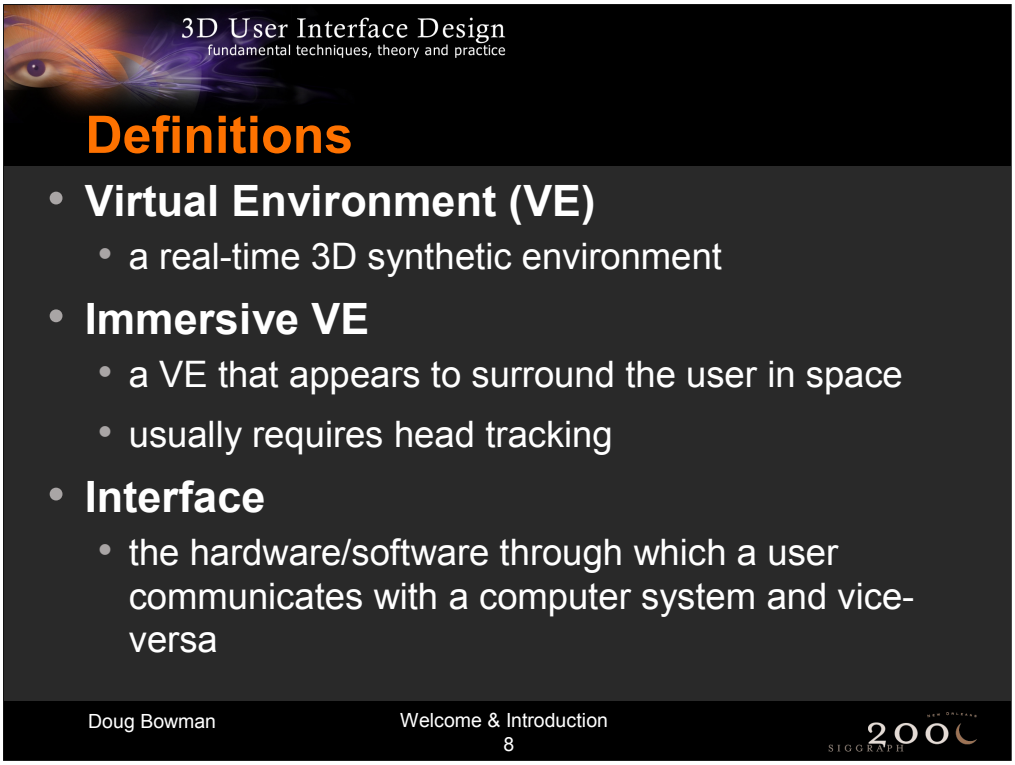
Finally, we include in each lecture some “take-home” messages which can be immediately applied to 3D interaction problems. These take the form of guidelines (presented in white italicized text) and myths (presented in orange italicized text).



Unfortunately, one day is not enough to cover the broad range of topics related to 3D interaction. We will mostly limit ourselves to talking about interfaces and interaction techniques for virtual environments (VEs) – that is, to applications which display 3D information and which require 3D input/interaction.

We will not have time to talk about augmented reality (systems where computer-generated graphics are overlaid on the physical world) or desktop 3D applications (systems where the displayed information is 3D, but interaction is mainly 2D or 2.5D). However, many of the general principles we discuss are applicable in these areas.

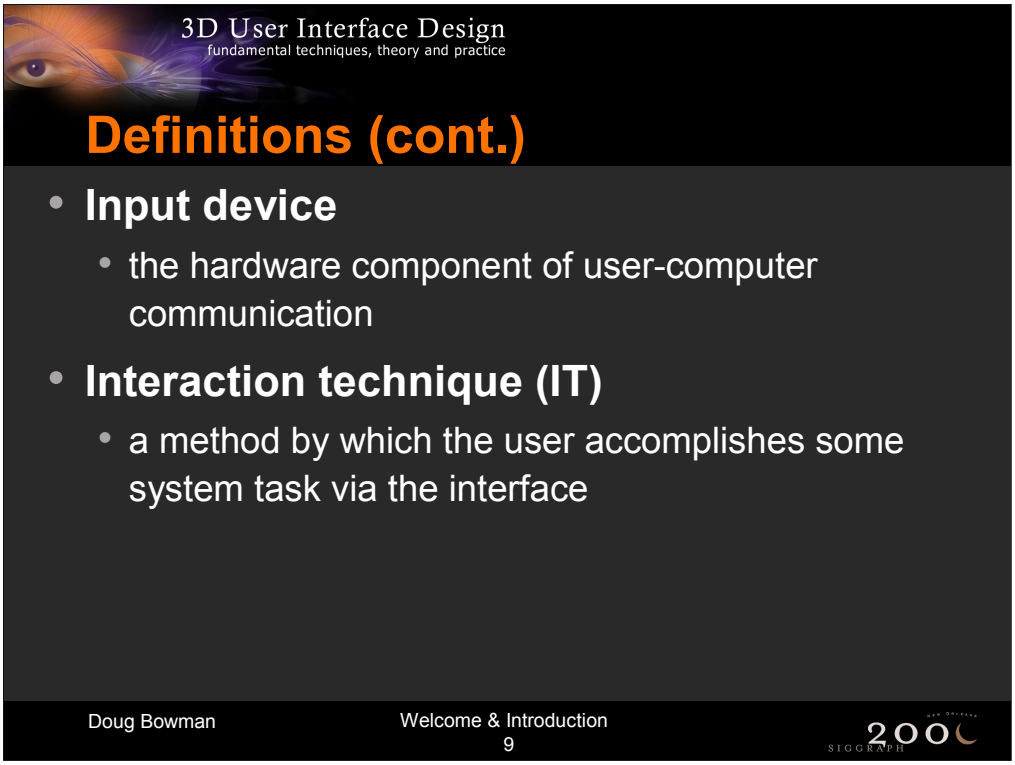
Also, we will not propose or discuss any standards for 3D interfaces (it's not clear that such standards are useful or needed at this point), or software packages that allow the development of 3D interfaces (there are very few good ones).



The definition of a VE is disputed widely, but we'll be talking about 3D computer-generated environments that are rendered in real-time (of course, real-time is also a vague term – we mean that the system updates fast enough and has minimal lag so that good interactivity is maintained).

An immersive VE uses head tracking to create the illusion that the environment surrounds the user. This *can* create immersion, or presence, where the user feels as if he's actually in the virtual world, but this is not always true. Research is ongoing into the causes of immersion, and a surrounding display is only one of many factors that contribute.

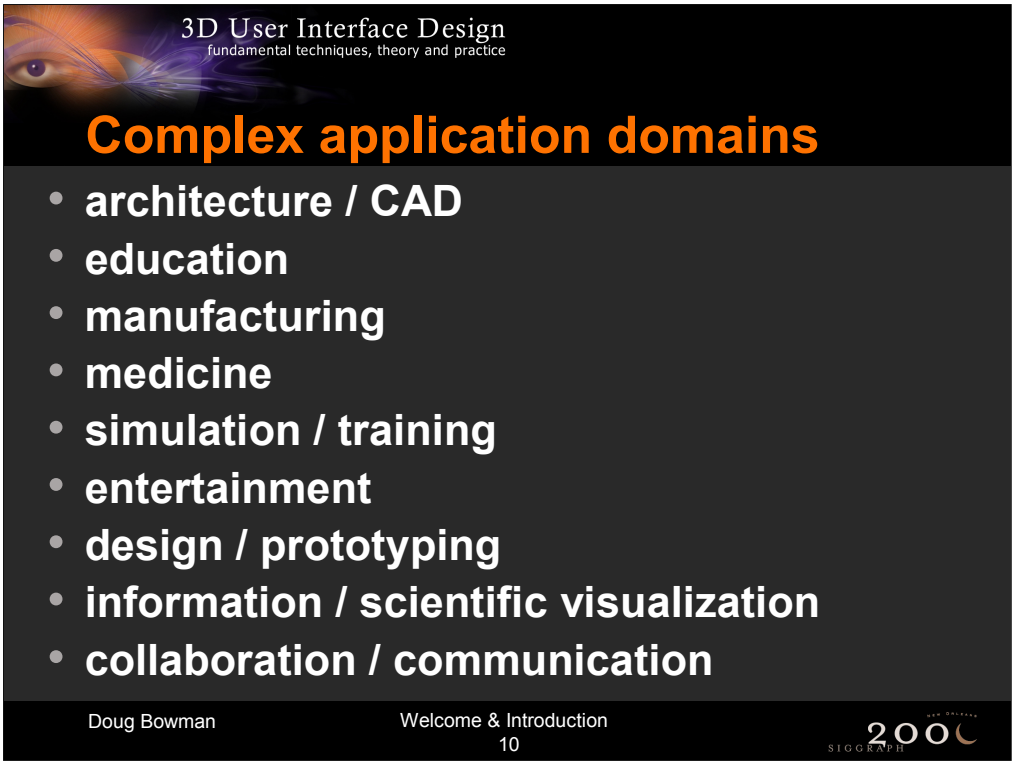
The user interface is simply a communications medium. The user communicates to the system via input of various types. The system presents information to the user via displays/output. 3D interfaces have the potential to greatly increase the bandwidth of the communications from the user to the system and from the system to the user.



An input device is simply some piece of hardware that is used to communicate with the system (e.g. mouse, keyboard, cyberglove, stylus, touch screen, etc.). We will discuss 3D input devices in the first lecture.

An interaction technique is part of the user interface (UI). It is a method that allows the user to perform some task in the system, and it includes both hardware (input device) and software components.

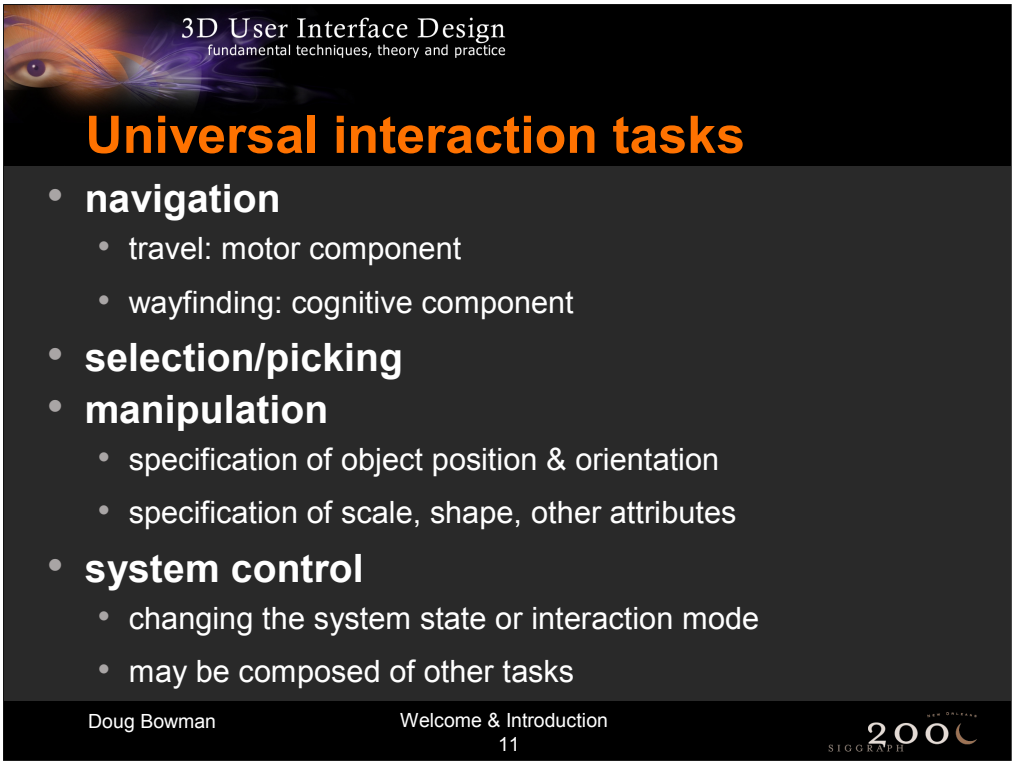
There is often confusion regarding these two terms. It is sometimes assumed that the choice of input device determines the interaction technique. With very special-purpose input devices, this is mostly true. For example, a touch screen device only makes sense if the item directly beneath the user's finger is selected. However, with most general-purpose input devices, many interaction techniques can be developed. Take the mouse for example. The buttons have a default action, but this is usually customizable by the user. Dragging an icon using the mouse does not always work the same way – things like gravity can be used so that the icon is drawn to its likely targets. Mice can be used to do simple 2D gestures. All of these examples show that the same input device can be used for many interaction techniques, and this is even more true for 3D interfaces. The reverse is also true – a single interaction technique can often be implemented using many different input devices.



The topics we'll be covering in this course apply to a wide range of potential VE applications, all of which might have complex requirements for user interaction.

There's a distinction between the **frequency** of interaction and the **complexity** of interaction. The simplest VE (look around a virtual world from a stationary viewpoint) may have very frequent user interaction (changing the view orientation), but this interaction is very simple (just based on head movements). These applications may all require complex navigation, object manipulation, etc.

Why should these applications benefit from 3D/VE implementation? It goes back to the list of VE characteristics that we listed earlier. Many of these applications will be enhanced if the user feels immersed, or if the user can act in a natural manner, or if an immediate visualization of the results of an action is presented.



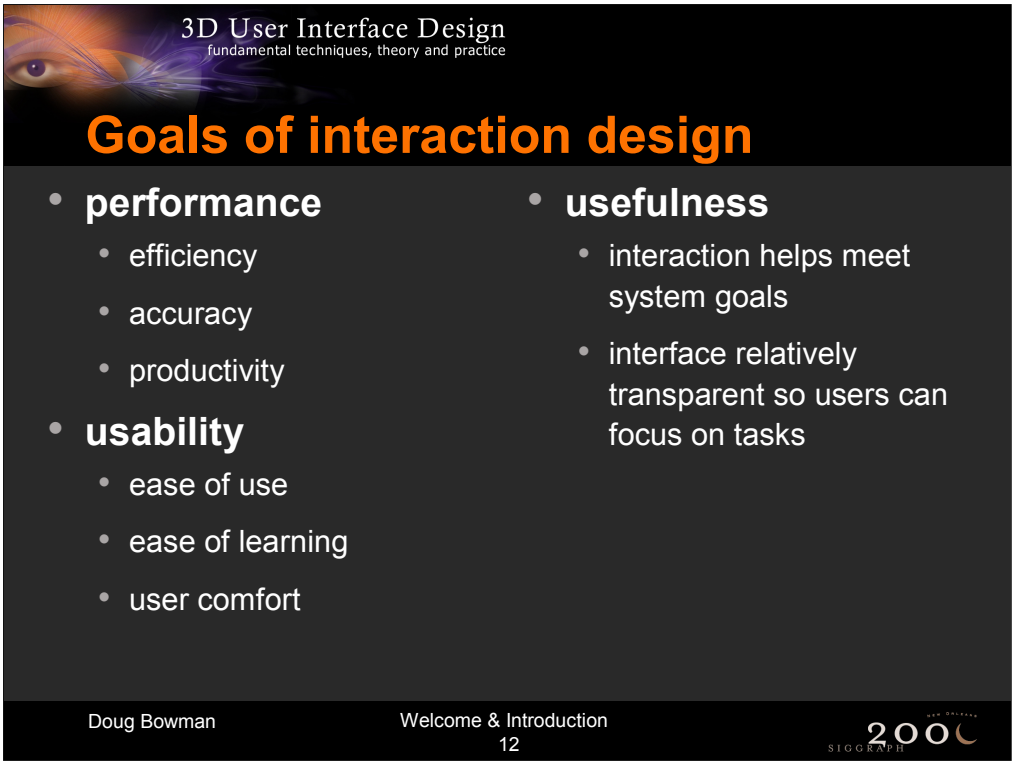
We'll be considering four "basic" or "universal" interaction tasks for 3D/VE applications. Obviously, there are other tasks which are specific to an application domain, but these are some basic building blocks that can often be combined to create a more complex task.

Navigation is the most common VE task, and is actually composed of two tasks. Travel is the motor component of navigation, and just refers to the physical movement from place to place. Wayfinding is the cognitive or decision-making component of navigation, and it asks the questions, "where am I?", "where do I want to go?", "how do I get there?", and so on.

Selection is simply the specification of an object or a set of objects for some purpose. Manipulation refers to the specification of object properties (most often position and orientation, but also other attributes). Selection and manipulation are often used together, but selection may be a stand-alone task. For example, the user may select an object in order to apply a command such as "delete" to that object.

System control is the task of changing the system state or the mode of interaction. This is usually done with some type of command to the system (either explicit or implicit). Examples in 2D systems include menus and command-line interfaces. It is often the case that a system control technique is composed of the other three tasks (e.g. a menu command involves selection), but it's also useful to consider it separately since special techniques have been developed for it and it is quite common.

These tasks will be covered in detail in the "interaction techniques" portion of the course.



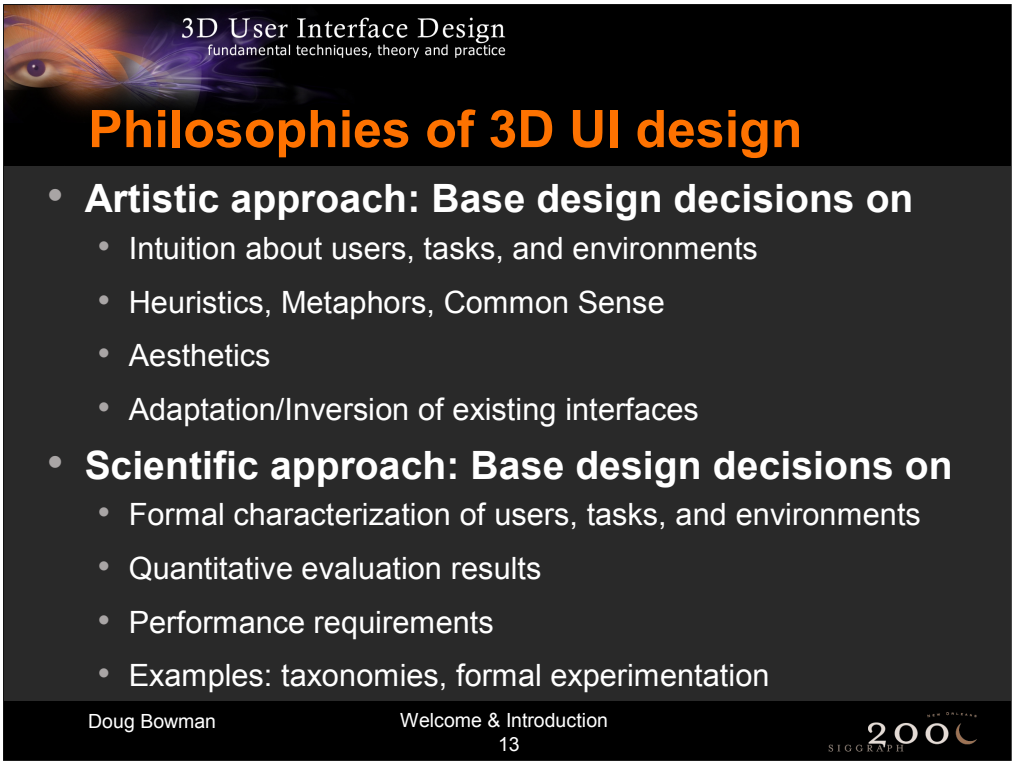
We will try to keep in mind as we discuss ways to accomplish all of these tasks that we want to design for performance, usability, and usefulness.

Performance relates to quantitative measures indicating how well the task is being done by the user and the system in cooperation. This includes standard metrics like efficiency and accuracy.

Usability refers to the ease of communicating the user's intentions to the system, and the qualitative experience of the user.

Usefulness implies that the system is actually helping the user perform work or meet his/her goals, without being hindered by the interface.

All three of these goals must be considered together, as all are essential. A system will not be used if users become frustrated after five minutes of usage (usability) even if it's been shown to aid the user in getting work done in a new way. A business will not adopt a system that is incredibly easy to use but decreases productivity (performance).

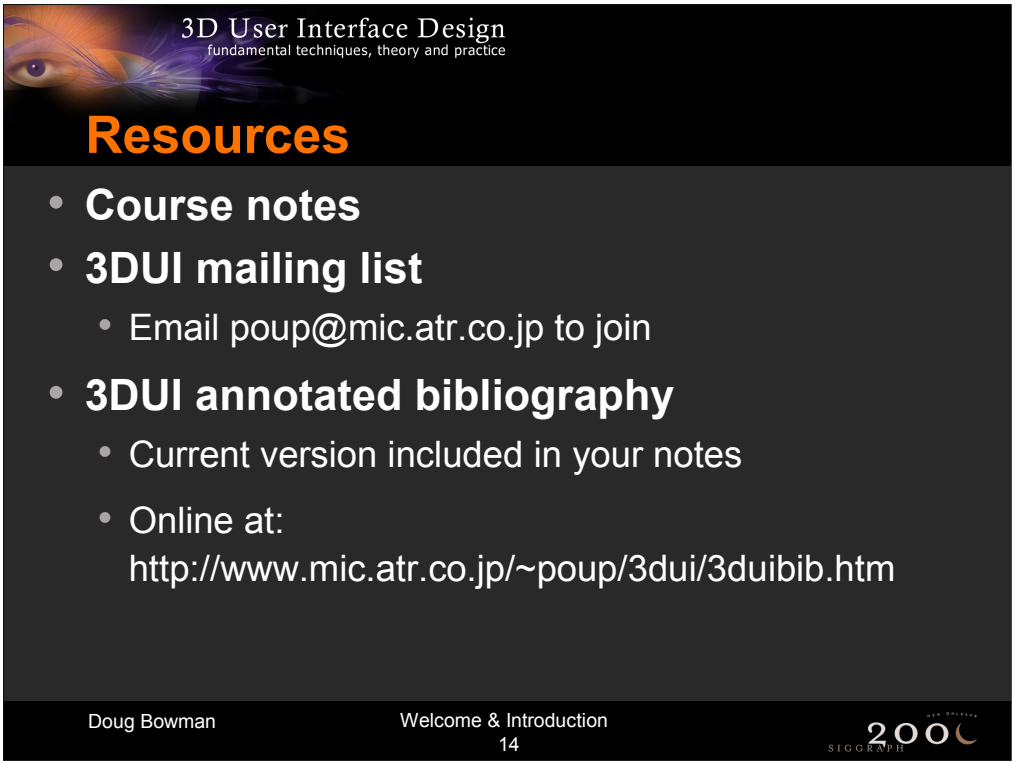


We'll also be thinking about about the philosophy behind the design of 3D user interfaces.

One philosophy is the "artistic" approach, which is based on intuition, common sense, heuristics, etc. This approach can be quite successful if well-grounded. The lecture on using 2D interfaces in a 3D world is an example of adapting existing interfaces, which can be considered part of this approach.

A second possible philosophy is the scientific approach, which attempts to formalize the design process by producing formal characterizations of the tasks, users, etc. involved; by doing quantitative experiments to determine the performance of interaction techniques, and so on.

Both of these approaches can lead to guidelines and principles that can be applied by developers. The artistic approach tends to be more innovative and novel, while the scientific approach tempers that innovation with real data, and reveals failures that can seed new innovations.



These course notes offer a great deal more information than we can possibly present during the one-day session at the conference. Wherever possible, we have attempted to expand upon the concepts addressed in the slides, provide references to published work, provide URLs for more information, etc.

The presenters of this course are also part of a large online community (the 3DUI mailing list) dedicated to the discussion of 3D interaction. We currently have over 75 members from more than 16 countries. If you would like to join, send an email to Ivan Poupyrev at poup@mic.atr.co.jp.

We have also included in your notes the current version of an annotated bibliography of 3D interfaces compiled from the members of the mailing list. You can access this bibliography online, and send in your own submissions or annotations.