



TECHNISCHE UNIVERSITÄT
CAROLO-WILHELMINA
ZU BRAUNSCHWEIG

Flexible User Interfaces For Mobile Devices

Oliver Bende

Advisor: Matthias Kranz

Seminar Ubiquitous Human Computer Interaction
Summer term 2007

Institute Of Operating Systems And Computer Networks
Technische Universität Braunschweig



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

- ✘ **User interfaces are one of the most important parts**
- ✘ **Difficult to develop because of the wide range of devices**
- ✘ **Flexible user interfaces cope with different capabilities and circumstances**
- ✘ **Are developed for many years**



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

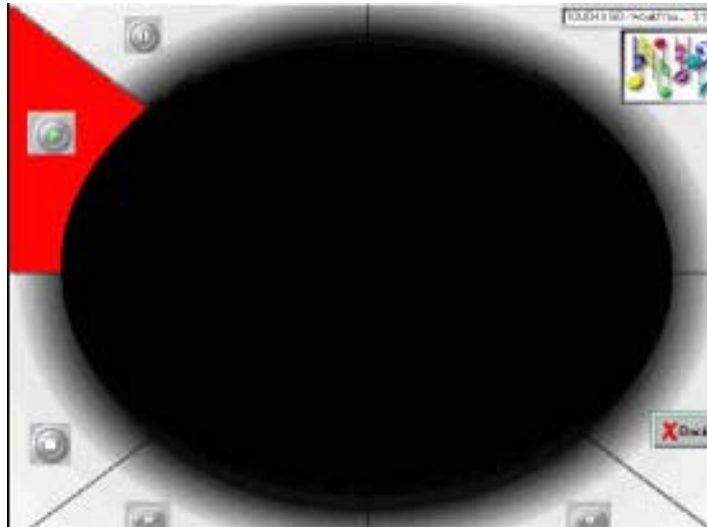
Context

Statement

- × **Different possibilities for output of information (Output)**
 - ▶ Audible, Displays
- × **Different possibilities for input of information (Input)**
 - ▶ Keyboards, pointing devices, speech, specific devices
- × **Internal conditions (calculating power, battery lifetime,...)**
 - ▶ Possibility to show a web interface instead of running application (Web)
- × **Context-awareness (Context)**
 - ▶ Adaptation of display to changing circumstances
- × **Short statement about actual situation**

× Different requests to interface

- ▶ HMD has other requests than other “usual” displays
 - wGUI for a HMD (Head Mounted Display)





Output methods: Usual displays

Introduction

Output

Input

Web

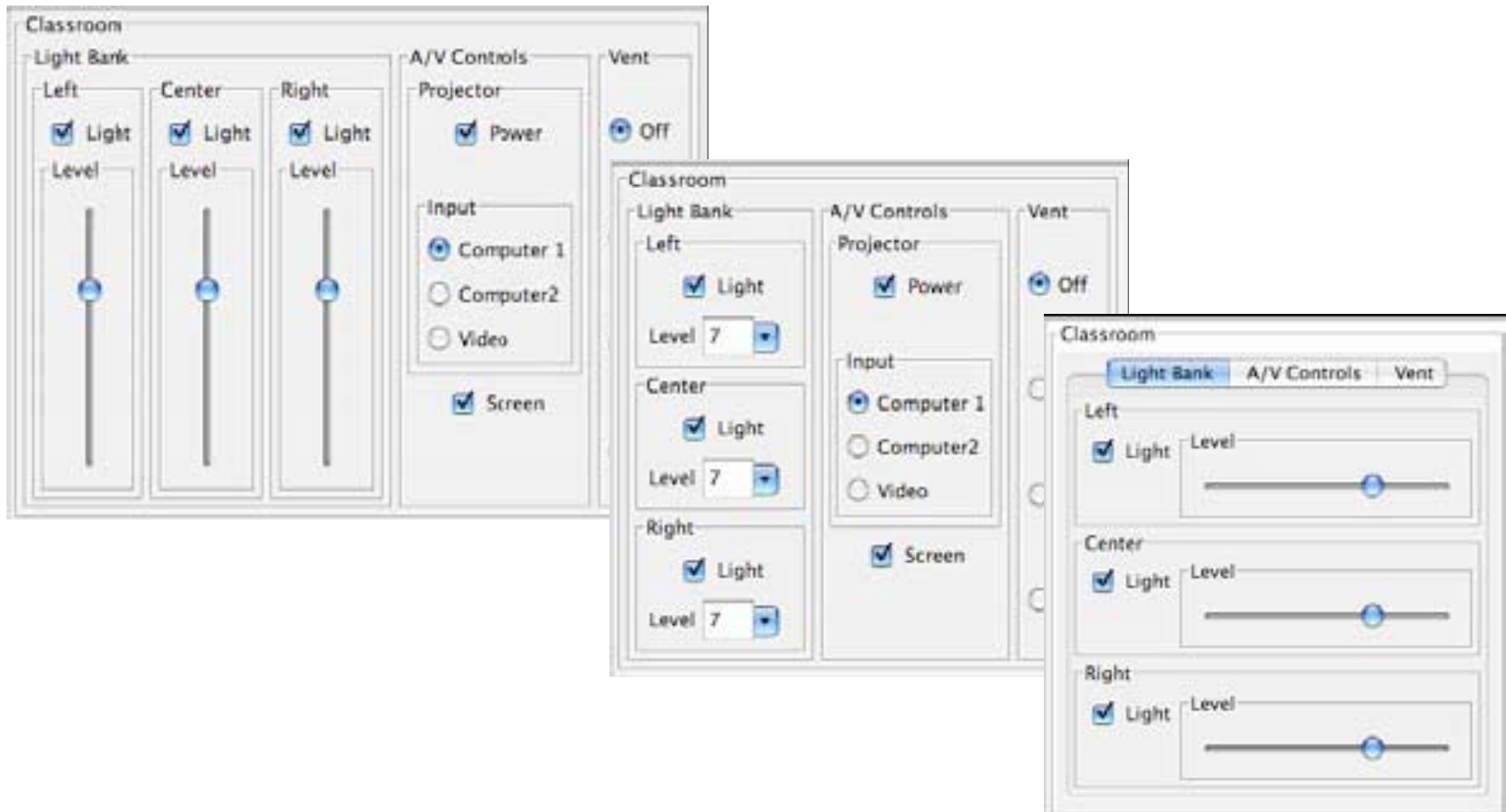
Context

Statement

- ✘ **One user interface for all devices inappropriate**
 - ▶ Tiny on desktop monitor or scrolling on mobile display
- ✘ **Adaptation to according display: Tasks**
 - ▶ Adapt to many different devices and circumstances
 - ▶ Never confuse the user (should look like the used one)
 - Decide what is important
- ✘ **Model-based and rule-based toolkits**
 - ▶ Model-based: abstract models
 - ▶ Rule-based: strict rules how interface should look

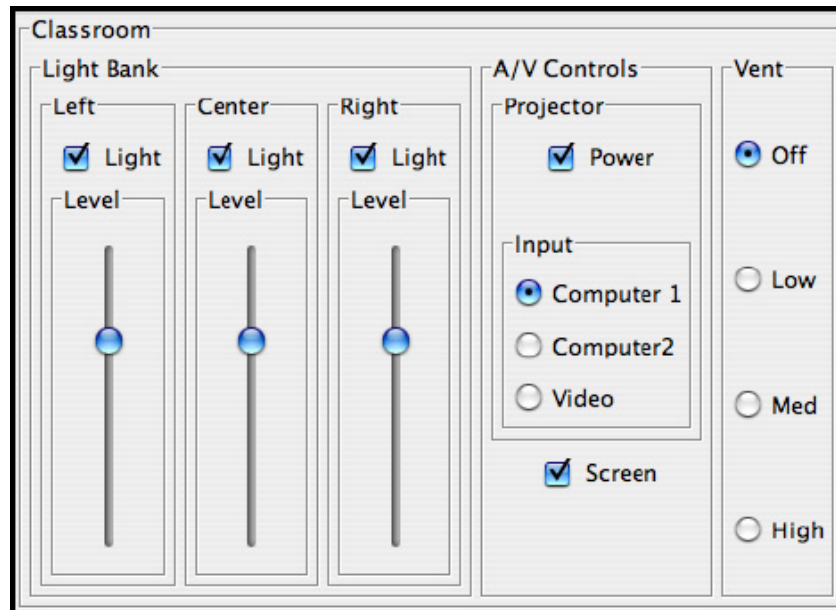
- × **Model-based**
 - ▶ Functional interface specification
 - ▶ Device-model describes capabilities
 - ▶ User-model (typical activities)
- × **Special algorithm to calculate user effort**
 - ▶ Cost for every possible interface
 - ▶ Branch and bound
 - ▶ Chooses the best possible interface
- × **Adapts itself to user actions at runtime**

✘ The same application rendered on different devices

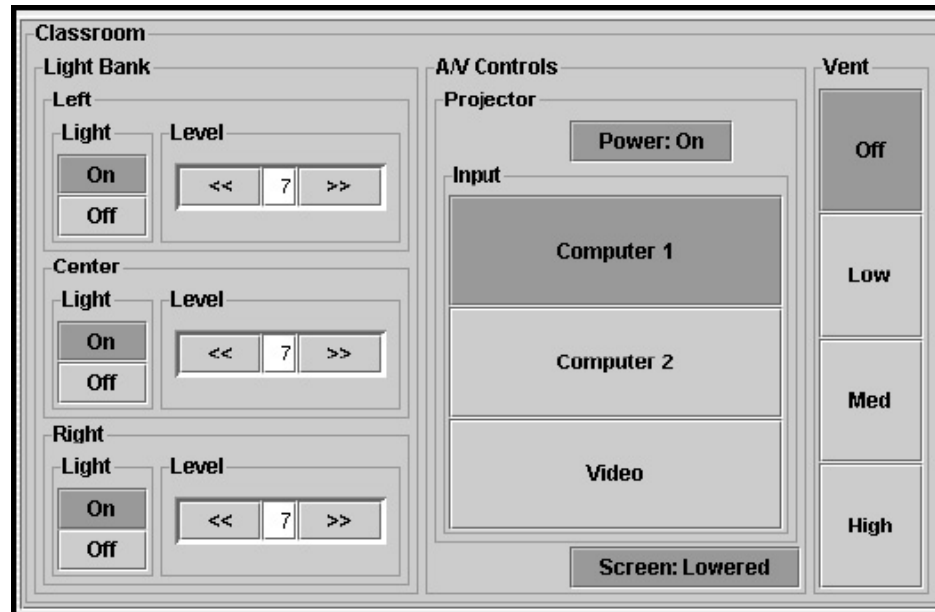


- × **Calculation very elaborate for processor**
- × **Remote rendering possible**
 - ▶ Automatic usage of “*solver server*”
 - ▶ Remote rendering service accelerates process
 - ▶ Network connection essential
- × **No support for context events**
- × **Enhancement: Factor in the cost to learn new interface**
 - ▶ Same application with new device

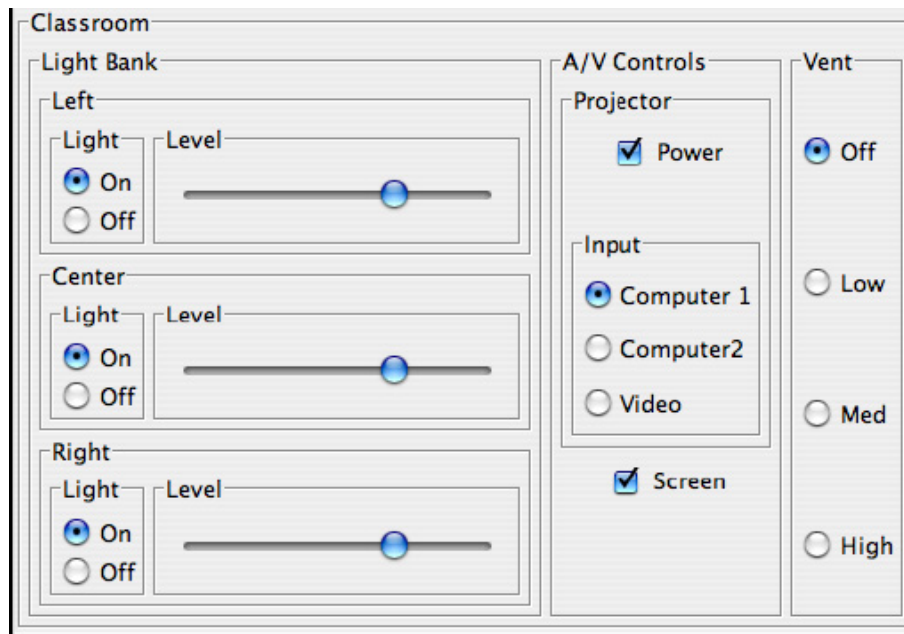
- ✘ **SUPPLE makes trade-off between optimization and similarity**



- ✘ SUPPLE makes trade-off between optimization and similarity

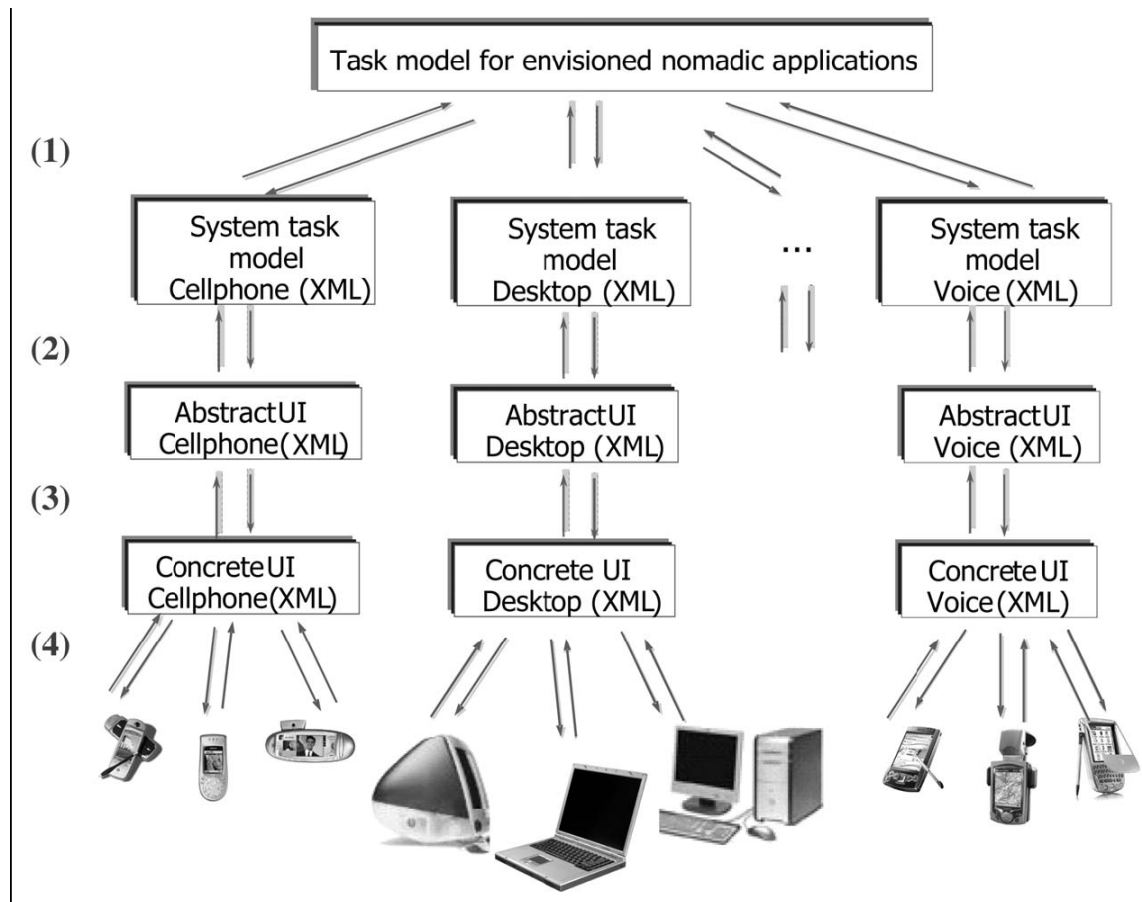


- ✘ **SUPPLE makes trade-off between optimization and similarity**



- × **Model-based tool**
- × **Different abstraction layers**
 - ▶ Top-down method
- × **XML-Based**
 - ▶ Possibility to import/export definitions
- × **Only flexible at generation, not at runtime**
- × **One task-model yields many interfaces**

✘ The „one model, many interfaces approach“





The Ubiquitous Interactor

Introduction

Output

Input

Web

Context

Statement

- ✗ **Creates device-independent user interfaces**
- ✗ **Gives the ability to control presentation of interface**
 - ▶ Important for branding purposes
- ✗ **Presentation information separated from user-service information**
- ✗ **Uses mappings and media resources**
 - ▶ Media resources: pictures or sounds
 - ▶ Mappings: link interactions to templates
- ✗ **Interface generation by server: Network connection required**



- × **Rule-based interface generator**
 - ▶ Specific knowledge about target-device is required
- × **Developed to access functions of appliances**
- × **Architecture has four parts**
 - ▶ Appliance adaptor
 - ▶ Specification language
 - ▶ Communication protocol
 - ▶ Interface generator



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

- × **Different possibilities for output of information (Output)**
 - ▶ Audible, Displays
- × **Different possibilities for input of information (Input)**
 - ▶ Keyboards, pointing devices, speech, specific devices
- × **Internal conditions (calculation power, battery lifetime,...)**
 - ▶ Possibility to show a web interface instead of running application (Web)
- × **Context-awareness (Context)**
 - ▶ Adaptation of display to changing circumstances
- × **Short statement about actual situation**



Input Methods

Introduction

Output

Input

Web

Context

Statement

- × **Users familiar with WIMP (Windows, Icons, Menus and Pointer)**
 - ▶ Mostly not possible for mobile devices
- × **Two popular input methods**
 - ▶ Four-way cursor
 - ▶ Touch screen
- × **Translations / adaptations**
 - ▶ Soft keyboard
 - ▶ Input Adaptor Tool
- × **Completely new methods**



Input Adapter Tool

Introduction

Output

Input

Web

Context

Statement

- × **Two problems to solve:**
 - ▶ User must be able to select any interactor
 - ▶ User must be able to control the selected interactor
- × **IAT adapts the running software to given input method**
- × **Manipulates the GUI**
 - ▶ Has to have control over the GUI
- × **Very general and applicable to all devices**



× Four input methods

- ▶ Pointer (no keyboard)
 - Suitable for most navigation
- ▶ Switch
- ▶ Speech
 - Cursor moves with regulation of voice pitch
 - Spatial: speak out grid regions
 - Jumping or scanning
- ▶ Keyboard (no pointer)
 - Tab-stop or direct mapping (jumping)

- ✗ Automatic addition of shortcuts
- ✗ Replacement of combo-boxes with text fields

The screenshot shows a web form titled "IAT Airline Form" with two sections: "Enter Purchaser Information" and "Enter Flight Information".

Enter Purchaser Information:

- First Name:
- Middle Name:
- Last Name:
- Gender: Male Female
- SSN: - -
- Address 1:
- Address 2:
- City: State: Zip: -
- Phone: - - Email:

Enter Flight Information:

- Depart: Arrive: Return:
- Depart Date: Arrive Date: Return Date:
- Submit!

The screenshot shows the same "IAT Airline Form" but with several modifications:

- First Name, Middle Name, and Last Name fields now have blue letters (g, h, k) above them, indicating shortcuts.
- Gender, SSN, Address 1, Address 2, Phone, and Email fields remain the same.
- City, State, and Zip fields now have blue letters (c, w, l, r) above them, indicating shortcuts.
- Depart, Arrive, and Return fields now have red backgrounds and blue letters (A, s, z) above them, indicating shortcuts.
- Depart Date, Arrive Date, and Return Date fields now have red backgrounds and blue letters (A, s, z) above them, indicating shortcuts.
- A dropdown menu is visible for the "Depart" field, showing options: "Albert Lea MH", "Albuquerque", "Alma NE", and "Alturas CA".
- Submit!

- ✗ **Refine of touch screen pointing device (stylus)**
- ✗ **Supports the recognition of wrist pressure**
 - ▶ Sensor under the bottom right screen of a PDA
- ✗ **User lays down the hand for editing**
- ✗ **User lifts hand for navigation**
- ✗ **Very specific – not for use with other methods / devices**





Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

- ✘ **Different possibilities for output of information (Output)**
 - ▶ Audible, Displays
- ✘ **Different possibilities for input of information (Input)**
 - ▶ Keyboards, pointing devices, speech, specific devices
- ✘ **Internal conditions (calculation power, battery lifetime,...)**
 - ▶ Possibility to show a web interface instead of running application (Web)
- ✘ **Context-awareness (Context)**
 - ▶ Adaptation of display to changing circumstances
- ✘ **Short statement about actual situation**



- ✘ **Displaying web pages on mobile devices**
 - ▶ No need to install additional software
- ✘ **Common languages are used (Java, XML,...)**
 - ▶ Many people can handle toolkits
- ✘ **Adaptive User Interface Toolkit (AUIT)**
 - ▶ Includes Java Swing based design tool
 - ▶ Combination of XML and Java Server Pages (JSP)
 - Other languages are possible (prototype)
 - ▶ Specifications stored in XML format
- ✘ **Drawback: Server connection essential**



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

- ✘ **Different possibilities for output of information (Output)**
 - ▶ Audible, Displays
- ✘ **Different possibilities for input of information (Input)**
 - ▶ Keyboards, pointing devices, speech, specific devices
- ✘ **Internal conditions (calculation power, battery lifetime,...)**
 - ▶ Possibility to show a web interface instead of running application (Web)
- ✘ **Context-awareness (Context)**
 - ▶ Adaptation of display to changing circumstances
- ✘ **Short statement about actual situation**



- ✘ **User interface which adapts the whole interface**
- ✘ **E.g. Sulawesi is able to adapt input and output**
 - ▶ Detects movement:
 - switches from pointing device to speech input and from display to audio output
 - ▶ Adapts on commands:
 - “tell me the time” or “show me the time”

- ✘ **User interface which only adapts its display**
- ✘ **Developed for Symbian mobile phones**
- ✘ **Adapts operating system GUI to different devices**
- ✘ **Location-based context-awareness (not yet implemented)**
 - ▶ Locate device (Cell ID or GPS)
 - ▶ Adapting interface to the users task
 - work / home
 - ▶ Copes with local context (shop / cinema poster)
 - Active / passive
- ✘ **Display-adaptation already implemented**



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

- × **Different possibilities for output of information (Output)**
 - ▶ Audible, Displays
- × **Different possibilities for input of information (Input)**
 - ▶ Keyboards, pointing devices, speech, specific devices
- × **Internal conditions (calculation power, battery lifetime,...)**
 - ▶ Possibility to show a web interface instead of running application (Web)
- × **Context-awareness (Context)**
 - ▶ Adaptation of display to changing circumstances
- × **Short statement about actual situation**



Statement

Introduction

Output

Input

Web

Context

Statement

- ✘ **Server-based systems only applicable for concerns with intranet**
 - ▶ Online-services too expensive and unstable
- ✘ **Some features already implemented**
- ✘ **Separated projects, prototypes and researches**
- ✘ **“Designing flexible and dynamically configurable user interfaces (UIs) is difficult and it is unlikely that the developer will come up with a solution of all problems for all users. “**

(Bogdan Dorohonceanu, Boi Sletterink, Ivan Marsic: A Novel User Interface for Group Collaboration)



Flexible User Interfaces For Mobile Devices

Introduction

Output

Input

Web

Context

Statement

Questions ?