POSIX

Ricardo.Sanz@etsii.upm.es

Based on Slides from Ken Frazer

Curso 2005-2006

Open Systems Standards

- Efforts have been underway for a number of years in the form of "**Open Systems**" standards

- Generally, an open systems standard means any vendor can build hardware and/or software to meet the standard and **expect interoperability**

POSIX To The Rescue

○ A major player is the **IEEE Portable Operating System Interface** (POSIX)

   ● Also known as IEEE1003 (or 1003.x) and ISO/IEC 9945

○ POSIX, however, refers only to software interfaces (not enough for whole portability assurance)

Portability

○ There are two main kinds of portability

  ● Source code

  ● Binary

○ Of the two, binary is much more difficult to achieve because it puts constraints on the machine hardware architecture and instruction set

Available Standards

- Efforts on standards have therefore concentrated on developing standard interfaces for source code

- Open system standards for source code portability define interfaces for essential services
  - Process control and interprocess communication
  - File and directory services

The Standard to End All Standards

- Actually, no single standard provides all the functionality needed in modern computing systems, nor would we want it to

- IEEE 1003.x (POSIX) handles this but dividing the standards into smaller pieces, each concentrating on a specific area

IEEE 1003.x Standards

- 1003.0 POSIX Open Systems Environment
- **1003.1** System Application Programming Interface
- **1003.2** Shell and Utilities
- 1003.2a User Portability Extension
- **1003.4** Real-time Extensions
- 1003.5 Ada Language Binding
- 1003.6 Security Extensions

IEEE 1003.x Standards

- 1003.7 System Administration
- 1003.8 Transparent (Network) File Access
- 1003.9 Fortran Interface
- 1003.12 Protocol-Independent Network Interface
- 1003.15 Batch Scheduling
- 1003.17 Namespace and Directory Service

- **1003.21** Real-time networking (???)

Who?

- The IEEE Computer Society an Open Group **Portable Application Standards Committee** (PASC)
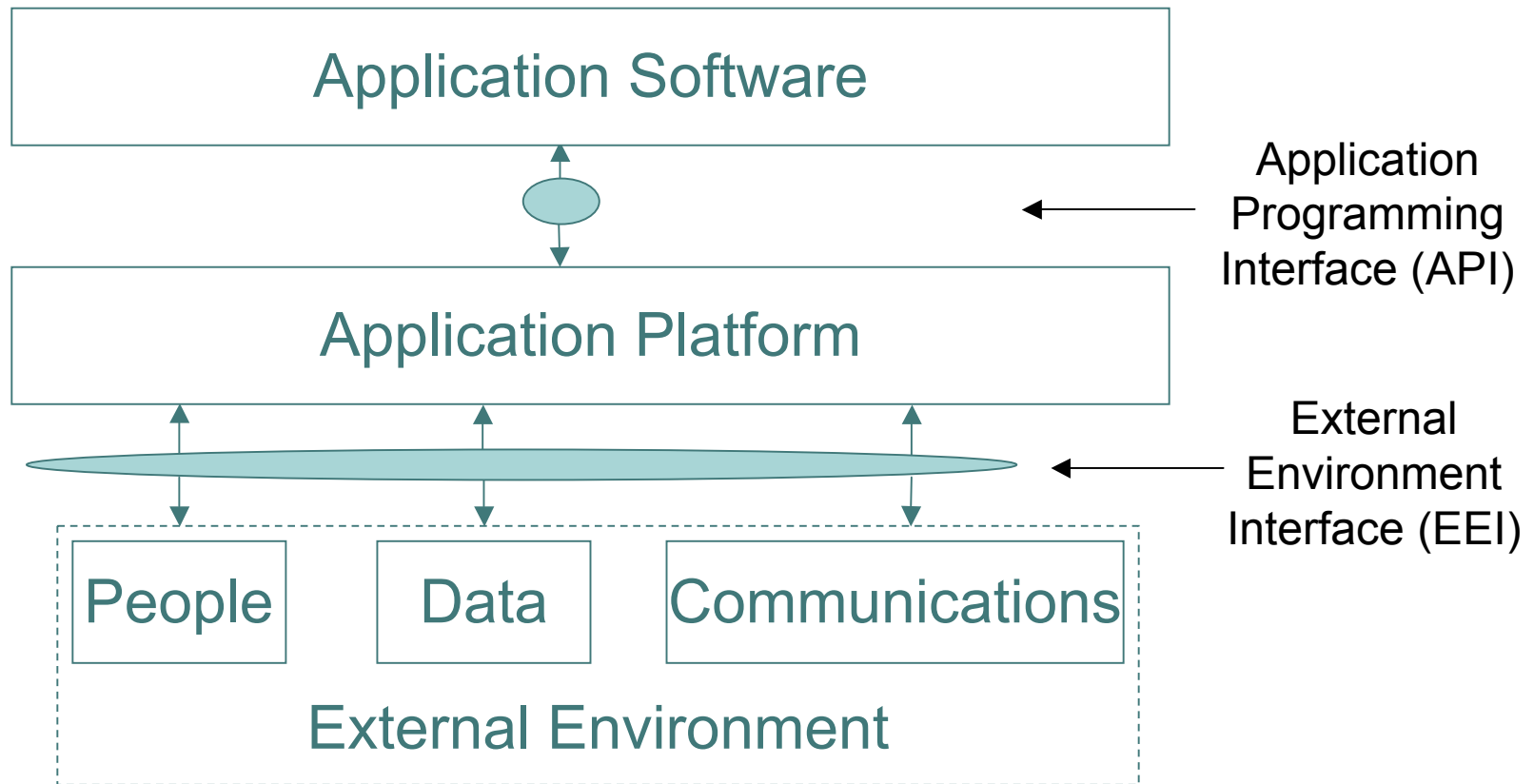
- www.pasc.org

Two Types of Standards

○ There are two types of standards specified in the POSIX model

  ● The application program interface (API)

  ● The external environment interface (EEI)

POSIX Reference Model

Application Software

Application Platform

People Data Communications

External Environment

Application Programming Interface (API)

External Environment Interface (EEI)

Application Programming

○ The APIs generally are the function calls used by the application to request services from the application platform

- i.e. calls to the computer on which the application programming is running and its operating system

External Environment

- The external environment generally refers to external entities which the application platform exchanges information

  - Human end-user

  - Hard copy documentation

  - Physical devices and services

    - Printers, displays, disk drives, networks
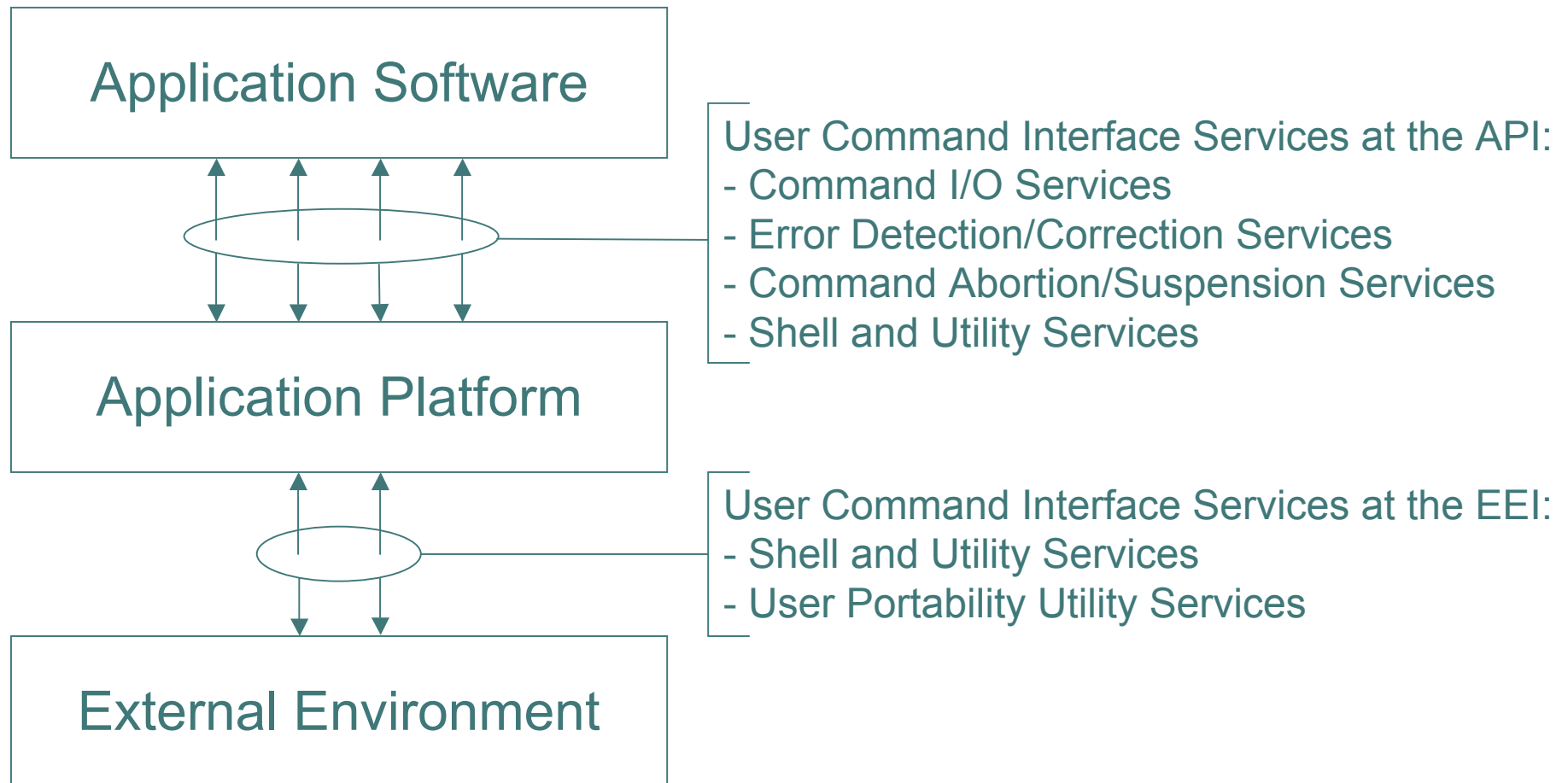
POSIX - General Objectives

- Application S/W portability at the source-code level
- Data portability
- Application S/W interoperability and application platform interoperability
- User portability
- Accommodation of standards
- Accommodation of new technology
- Application platform scalability
- Distributed system scalability
- Implementation transparency

Services

- Services in the POSIX specification fall into three categories
  - API, EEI and API/EEI
- These services cover three major aspects of portability and interoperability
  - Basic functionality for general usefulness
  - Ability to move applications between platforms
  - User portability, i.e. availability of the same command interface across platforms

User Command Interface Services

Application Software

User Command Interface Services at the API:
- Command I/O Services
- Error Detection/Correction Services
- Command Abortion/Suspension Services
- Shell and Utility Services

Application Platform

User Command Interface Services at the EEI:
- Shell and Utility Services
- User Portability Utility Services

External Environment

POSIX API

○ **System services**
- including core system and language services

○ **Communications services**

○ **Information services**
- including database, data interchange, and transaction processing services

○ **Human/Computer Interaction services**
- including user command interface, character-based UI, windowing UI, graphics, and application development support services

POSIX API Services

- Services
  - Standard File and Terminal I/O
  - Files and Directories
  - Processes
  - Obtaining Information at Run-time
  - Terminal I/O

- **More detail in Lesson 12**

API and EEI Services

○ The API should be able to access all the services available to the user at the EEI

- Provide input to a command and access the output of the command when necessary

- Obtain additional data or commands from a file

- Detect and correct errors as a command is executed

- Abort or suspend a command as it is executing

API and EEI Services

○ It is important to have the ability to create script files that are combinations of commands

○ The scripting language must contain the following services:

- Execute commands conditionally
- Execute commands repeatedly
- Display output
- Prompt the user for input
- Execute commands and obtain error information

EEI Services Purpose

○ On a traditional system, these capabilities are implemented by providing interactive commands entered via a keyboard.

○ As graphical UIs evolve, these commands may also be implemented by clicking a mouse, by a touch screen, a tablet, or other input device

EEI Services

- Capture the output of a command or application into a file
- Redirect the input for a command from a file
- Direct the output of a command to be used as input to another command
- Execute applications
- Get online help for commands or applications

EEI Services

- Manipulate file contents
  - Cut, paste, concatenate, convert, sort, reformat, compare, search for regular expressions
- Edit files
  - Interactive editors, batch or "stream" editors
- Display files
  - Pausing when necessary
  - Display only selected ranges of files

EEI Services

- Manipulate files
  - Create, delete, rename, move, copy
- Print files
- Perform network functions
  - Transfer files
  - Execute commands remotely
  - Print files remotely
  - Etc.

EEI Services

- Perform batch processing
  - Create and manage batch queues
  - Submit, terminate, and get status of jobs
  - Retrieve output
- Manipulate and display directories
  - Create, delete, display, and destroy (delete a directory and all its subdirectories and files)
- Control file and directory permissions

EEI Services

- Communicate with other users
  - Send and receive electronic mail
  - Interact online
- Interrogate system information
  - Currently logged on users
  - Other information about users
  - Currently executing system processes
  - File system status
  - Kernel, disk interface, and network status

EEI Services

○ Control the application execution environment

  ● Execute applications in the background

  ● Abort applications running in the foreground or background

  ● Suspend an application

  ● Move an application running in the foreground to the background

○ Schedule commands for periodic execution

EEI Services

- Control the input equipment of the user, such as a terminal or graphical user interface

- Manage local environment and configuration

- Query local environment and configuration

- Configure an environment for an international locale

POSIX Shell and Utilities

○ IEEE 1003.2/2a provides two classes of service

- A command set that provides access to a specific set of services and utilities

- Additional, optional utilities and features that promote the portability of users and programmers, in addition to applications, across conforming systems

  - The User Portability Options

POSIX Standards Impact

- Most system vendors are now conforming to POSIX standards (specifically IEEE 1003.1)
  - Even Microsoft provides a set of POSIX utilities for the MS Windows platform
- Linux has a mostly POSIX API/EEI
- POSIX is no longer a market differentiator; it is expected that a UNIX product should comply with it (more after the Single UNIX Specification)

Current Specification

THE Open GROUP   ISO IEC   IEEE

Rationale

IEEE Std 1003.1,

ISO/IEC 9945

The Open Group Base
Specifications Issue 6

System Interfaces   Commands   Definitions

The Core of the
Single UNIX Specification V3

Current Specification

- After joining efforts (IEEE, Open Group and ISO/IEC) a unified specification has been produced:
  - Single UNIX Specification
  - IEEE POSIX.1 2001
- Available on-line at the Open Group
- Available at IEEE:

IEEE Std 1003.1-2004 IEEE Standard for Information Technology— Portable Operating System Interface(POSIX®)— Part 1: System Application Program Interface (API)

Online Availability

- See the last spec at:

  http://www.unix.org/version3/ieee_std.html

- **IEEE Std 1003.1, 2004 Edition**