

Principios del diseño de smaltalk

GENERAL

Dominio Personal: Si un sistema es para servir al espíritu creativo, debe ser completamente entendible para un individuo solitario.

Buen Diseño: Un sistema debería ser construido con un mínimo conjunto de partes no modificables; esas partes debieran ser tan generales como sea posible; y todas las partes del sistema deberían estar mantenidas en un esquema uniforme.

LENGUAJE

Alcance: El diseño de un lenguaje para usar computadoras debe tratar con modelos internos, medios externos, y con la interacción entre ellos tanto en el humano como en la computadora.

Propósito del lenguaje: Proveer un esquema para la comunicación.

OBJETOS QUE SE COMUNICAN

Objetos: Un lenguaje de computación debe soportar el concepto de "objeto" y proveer una manera uniforme de referirse a los objetos de universo.

Administración del Almacenamiento: para ser auténticamente "Orientado a Objetos" un sistema debe proveer administración automática del almacenamiento.

Mensajes: La computación debería ser vista como una capacidad intrínseca de los objetos que pueden ser invocados uniformemente enviándoles mensajes.

Metáfora Uniforme: Un lenguaje debería ser diseñado alrededor de una metáfora poderosa que pueda ser aplicada uniformemente en todas las áreas.

ORGANIZACIÓN

Modularidad: Ningún componente en un sistema complejo debería depender de los detalles internos de ningún otro componente.

Polimorfismo: Un programa sólo debería especificar el comportamiento esperado de los objetos, no su representación.

Clasificación: Un lenguaje debe proveer un medio para clasificar objetos similares, y para agregar nuevas clases de objetos en pie de igualdad con las clases centrales del sistema.

Factorización: Cada componente independiente de un sistema sólo debería aparecer en un sólo lugar.

Reaprovechamiento: Cuando un sistema está bien factorizado, un gran reaprovechamiento está disponible tanto para los usuarios como para los implementadores.

Máquina Virtual: Una especificación de máquina virtual establece un marco para la aplicación de tecnología.

INTERFAZ DEL USUARIO

Principio Reactivo: Cada componente accesible al usuario debería ser capaz de presentarse de una manera entendible para ser observado y manipulado.

Sistema Operativo: Un sistema operativo es una colección de cosas que no encajan dentro de un lenguaje. No debería existir.

TRABAJO FUTURO

Selección Natural: Los lenguajes y sistemas que son de buen diseño persistirán, sólo para ser reemplazados por otros mejores.

APARTE (NOTAS)

La máquina virtual de Smalltalk establece un modelo orientado a objetos para el almacenamiento, un modelo orientado a mensajes para el procesamiento, y un modelo de bitmap (mapa de bits) para el despliegue visual de información.

Smalltalk no tiene "sistema operativo" como tal. Las operaciones primitivas necesarias, como leer una página del disco, son incorporadas como métodos primitivos en respuesta a mensajes Smalltalk normales.

Todos los gráficos en Smalltalk se hacen con una sola operación primitiva.

Todas las clases tienen como superclase a la clase objeto.

En la práctica la referenciación uniforme (lo que da identidad a los objetos) se obtiene asociando un entero diferente a cada objeto del sistema.