CSE 303: Concepts and Tools for Software Development

Dan Grossman
Spring 2005
Lecture 22—Generics, function pointers, void*
Where are We

What I thought today was: version-control, e.g., cvs

What we better do based on questions: interfaces, generics, function-pointers, void*, passing pointers (to pointers).

A “new” approach: Analogous code in Java and C, starting with Java.
Very-High-Level Points

• Flexible “interfaces” often involve function-pointers.
  – You need “environments” so the function-pointers can have the data they need.

• Object and void* are analogous
  – And poor substitutes for type variables (Java 1.5) or arguably templates (C++) because you have to cast a lot.

• Homework 5/6 has one technique not in the example: function pointers that mutate their environment to “remember where they are”.