



LOP/DSL

Language Oriented Programming & Domain Specific Languages



Sioux

October 2009

Language Oriented Programming - Characteristics

- Designing a System Through Multiple Domain Specific Languages
- Is about organizing the development of large software structures
- Leading to a different structure for the finished product

Advantages

- Separation of Concerns
- High Development Productivity
- Highly Maintainable Design
- Highly Portable Design
- Opportunities for Reuse
- User Enhanceable System

Separation of Concerns

- Complete separation of concerns between
 - design issues, which are addressed in a domain specific language and
 - implementation issues, which are addressed in the implementation of the language

- With a problem-specific very high level language, a few lines of code are sufficient to implement highly complex functions
- The language implementation is also kept small since only those features which are relevant to the particular problem domain need to be implemented

Highly Maintainable Design

- Most important factor affecting maintainability: size of the software (lines of code)
- The small total size of a system produced by LOP implies high maintainability
- Major functions of the system are implemented as a few lines of code in an appropriate language
 - Bug fixing and enhancements are easy
 - Reduced chance of unexpected interactions

Highly Portable Design

- Porting to other language and/or operating system becomes greatly simplified:
 - only the middle language needs to be re-implemented on the new machine

- Programmers are heavily restricted by programming infrastructure
 - Languages
 - Environments
- Regaining freedom
 - Reducing dependencies
Example: Java to abstract operating system



The Problem

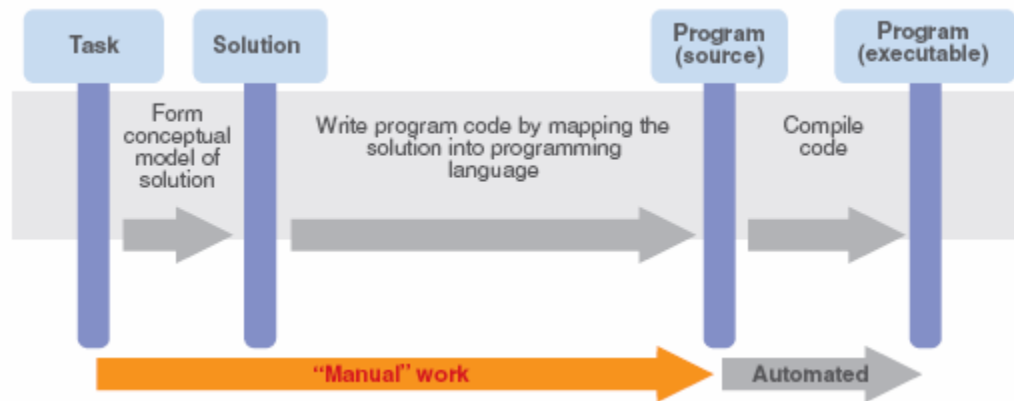
Regaining Freedom

- Reduce the level of dependency
 - Example: Java to reduce dependency on operating system

Think => Choose => Program

- **Think:** Create a conceptual model
- **Choose:** Choose a general purpose language (Java, C++, C#)
- **Program:** write the solution by performing a difficult mapping of the conceptual model into the programming language

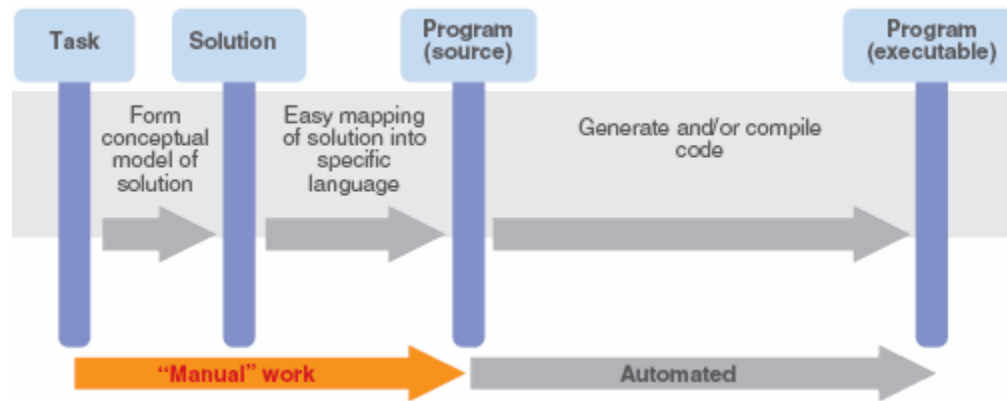
Mainstream Programming



Think => Choose => Create => Program

- **Think:** Create a conceptual model
- **Choose:** Select specialized DSLs for writing the solution
- **Create:** specialized DSLs (if not available)
- **Program:** relatively straightforward mapping of the conceptual model onto the DSLs

Language Oriented Programming



Mainstream Programming?

- Time delay to implement ideas
- Understanding and maintaining existing code
- Domain learning curve

Mainstream Programming

Today, ninety-nine percent of programmers think programming means writing out a set of instructions for the computer to follow.

We were taught that computers are modeled to the Turing machine, and so we “think” in terms of sets of instructions.

But this view of programming is flawed. It confuses the means of programming with the goal.

- Make it easy to create special domain-specific languages.
- Any unambiguous solution to some problem in some domain is a program.

What Is A Language in LOP

- In LOP there are three pillars
 - Structure
 - Defines its abstract
 - What concepts are supported
 - How they can be arranged
 - Editor
 - Defines its concrete syntax
 - How it should be rendered, and
 - Edited.
 - Semantics
 - Defines its behavior
 - How it should be interpreted and/or how it should be transformed into executable code.
- Other aspects: constraints, type systems, ...

What Is A Program Anyway?

- A **program** is any precisely defined **model** of a **solution** to some **problem** in some **domain**, expressed using **domain concepts**.
- We need to be able to easily create formal, precisely defined, domain-specific languages