我和OO的那些事儿

ACSIP

ZEMIN WANG

2020年8月

我和OO的那些事儿

- OO正在被淘汰？
- OO之历史，现状，及未来
- OOA的那些事儿
- OOD的那些事儿
- OOP的那些事儿

OO正在走向被淘汰？

OO正在被淘汰？

- 迅速崛起的AI系统和应用开发不需要OOP
- OOP不适用于大量开发的数据分析和BI应用
- 泛函编程比OOP更适合于现代系统和应用的编程
- 使用以构架（Framework）为主的，特别是CMS为基础的构架的大量应用开发，正在质疑和挑战OOP
- 质疑OO方法在现代程序系统和应用的适用性
- 以Agile为主流的系统开发过程和管理对OO的质疑和挑战
- 现代主流编程语言非OO-Only编程语言，均支持面向对象和过程编程
- …………

OO发展史看OO现状及未来

- 经历的OO发展史
- OOPSLA会议回顾
- 编程语言的分类

经历的OO发展史

始于80年代后期

- 上升期（ 1986 – 90年代中期 ）
- 发展期（ 90年代中期 – 2010前 ）
- 硕果期（ 2010 后 ）

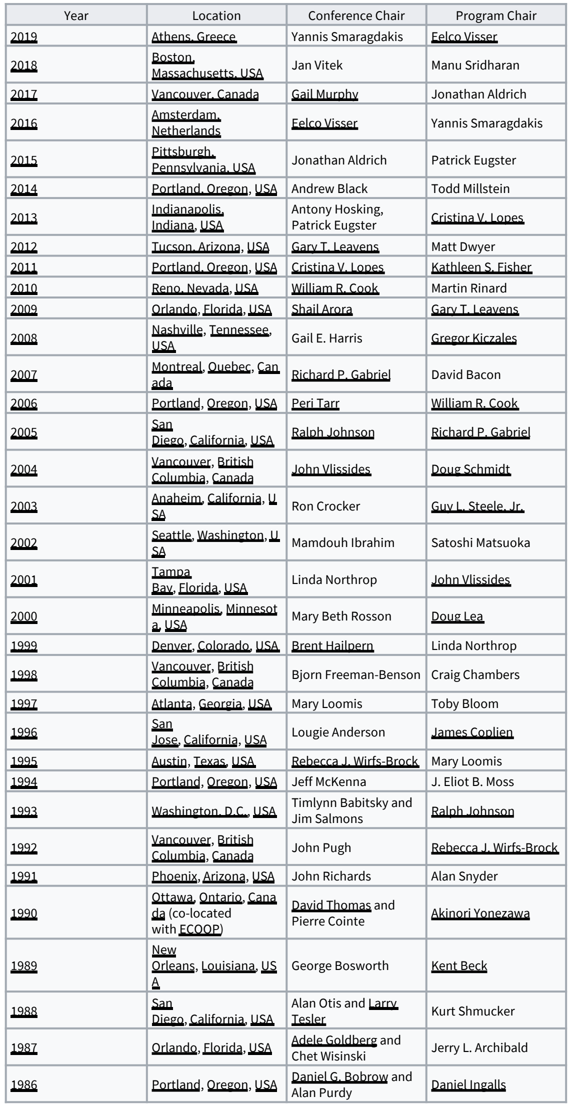
ACM SIGPLAN Research Conference OOPSLA

OO发展史，现状及未来

Since 2010, OOPSLA became a part of SPLASH Conference

ACM: Association for Computing Machinery
SIGPLAN: Special Interest Group for Programming Languages
OOPSLA: Object-Oriented Programming， Systems， Languages & Applications
SPLASH: Systems， Programming， Languages， and Applications： Software for Humanity

OOPSLA has been instrumental in helping object-oriented programming develop into a mainstream programming paradigm. It has also helped incubate a number of related topics, including design patterns, refactoring, aspect-oriented programming, model-driven engineering, agile software development, and domain specific languages.



Year	Location	Conference Chair	Program Chair
2019	Athens, Greece	Yannis Smaragdakis	Eelco Visser
2018	Boston, Massachusetts, USA	Jan Vitek	Manu Sridharan
2017	Vancouver, Canada	Gail Murphy	Jonathan Aldrich
2016	Amsterdam, Netherlands	Eelco Visser	Yannis Smaragdakis
2015	Pittsburgh, Pennsylvania, USA	Jonathan Aldrich	Patrick Eugster
2014	Portland, Oregon, USA	Andrew Black	Todd Millstein
2013	Indianapolis, Indiana, USA	Antony Hosking, Patrick Eugster	Cristina V. Lopes
2012	Tucson, Arizona, USA	Gary T. Leavens	Matt Dwyer
2011	Portland, Oregon, USA	Cristina V. Lopes	Kathleen S. Fisher
2010	Reno, Nevada, USA	William R. Cook	Martin Rinard
2009	Orlando, Florida, USA	Shail Arora	Gary T. Leavens
2008	Nashville, Tennessee, USA	Gail E. Harris	Gregor Kiczales
2007	Montreal, Quebec, Canada	Richard P. Gabriel	David Bacon
2006	Portland, Oregon, USA	Peri Tarr	William R. Cook
2005	San Diego, California, USA	Ralph Johnson	Richard P. Gabriel
2004	Vancouver, British Columbia, Canada	John Vlissides	Doug Schmidt
2003	Anaheim, California, USA	Ron Crocker	Guy L. Steele, Jr.
2002	Seattle, Washington, USA	Mamdouh Ibrahim	Satoshi Matsuoka
2001	Tampa Bay, Florida, USA	Linda Northrop	John Vlissides
2000	Minneapolis, Minnesota, USA	Mary Beth Rosson	Doug Lea
1999	Denver, Colorado, USA	Brent Hailpern	Linda Northrop
1998	Vancouver, British Columbia, Canada	Bjorn Freeman-Benson	Craig Chambers
1997	Atlanta, Georgia, USA	Mary Loomis	Toby Bloom
1996	San Jose, California, USA	Lougie Anderson	James Coplien
1995	Austin, Texas, USA	Rebecca J. Wirfs-Brock	Mary Loomis
1994	Portland, Oregon, USA	Jeff McKenna	J. Eliot B. Moss
1993	Washington, D.C., USA	Timlynn Babitsky and Jim Salmons	Ralph Johnson
1992	Vancouver, British Columbia, Canada	John Pugh	Rebecca J. Wirfs-Brock
1991	Phoenix, Arizona, USA	John Richards	Alan Snyder
1990	Ottawa, Ontario, Canada (co-located with ECOOP)	David Thomas and Pierre Cointe	Akinori Yonezawa
1989	New Orleans, Louisiana, USA	George Bosworth	Kent Beck
1988	San Diego, California, USA	Alan Otis and Larry Tesler	Kurt Shmucker
1987	Orlando, Florida, USA	Adele Goldberg and Chet Wisinski	Jerry L. Archibald
1986	Portland, Oregon, USA	Daniel G. Bobrow and Alan Purdy	Daniel Ingalls

List of Programming Languages by Types

1 Array languages
2 Assembly languages
3 Authoring languages
4 Constraint programming languages
5 Command line interface languages
6 Compiled languages
7 Concurrent languages
8 Curly-bracket languages
9 Dataflow languages
10 Data-oriented languages
11 Decision table languages
12 Declarative languages
13 Embeddable languages
14 Educational languages
15 Esoteric languages
16 Extension languages
17 Fourth-generation languages
18 Functional languages
19 Hardware description languages
20 Imperative languages
21 Interactive mode languages
22 Interpreted languages
23 Iterative languages
24 Languages by memory management type
25 List-based languages – LISPs
26 Little languages
27 Logic-based languages
28 Machine languages
29 Macro languages
30 Metaprogramming languages
31 Multiparadigm languages
32 Numerical analysis
33 Non-English-based languages
34 Object-oriented class-based languages
35 Object-oriented prototype-based languages
36 Off-side rule languages
37 Procedural languages
38 Query languages
39 Reflective Language
40 Rule-based languages
41 Scripting languages
42 Stack-based languages
43 Synchronous languages
44 Shading languages
45 Syntax handling languages
46 System languages
47 Transformation languages
48 Visual languages
49 Wirth languages
50 XML-based languages

https://en.wikipedia.org/wiki/List_of_programming_languages_by_type

OO发展史，现状及未来

OO发展史，现状及未来

基本编程语言概念及案例

编程语言转换类型：

- 汇编 （ Assembly ）
- 编译（ Compiled ）
- 解释 （ Interpreted ）

过程编程语言：

- Algol
- BASIC
- PL/I
- FORTRAN
- PASCAL
- COBOL

纯面向对象语言：

- SmallTalk

逻辑编程语言：

- Prolog

Functional 编程语言：

- Scheme
- Common LISP

混血（ Multiparadigm ）编程语言：

- Java
- PHP
- Python
- Scala

解释执行编程语言：

- Ant
- APL
- BASIC
- DM
- Eiffel
- JavaScript
- LISP

对OO发展的个人看法

- 不是终结而是继续。 并和其他编程语言长期共存
- 对OO编程语言和方法的质疑，主要来自于对OO的理解。
- 历史反复证明，OO的发展是循环渐进的。 噪声总是存在，进步与妥协共存
- OO是思维方式(Paradigm)和方法学。 一个OO人，在解决问题的过程中，很容易激发和感觉到创造的魅力

OOA的那些事儿

OOA的那些事儿

- 面向对象分析案列一： Identity Verification for Health Card Reengineering Project（ MOHLTC 2002 ）

- 面向对象分析案列二： Gene Data Management System（ Gene DB for Aventis 2003)

OOD的那些事儿

OOD的那些事儿

- 面向对象设计案列一： Bank Account Redesign （ Canada Trust 1999 ）

- 面向对象设计案列二： Web Site Performance Measure and Monitoring) (OOCL 2001)

OOP的那些事儿

OOP的那些事儿

- 面向对象编程案例一： OOCL Shipping Logistic Management System （ OOCL 2001 ）
- 面向对象编程案例二： Government Telehone Directory Management System (GovPage for MGCS 2004)
- 面向对象编程案例三：Government IT Source FFE RFS Management System (CAC 2015)

Q & A