

FSE598 前沿计算技术

模块 1 计算思维 单元 2 工作流与可视化编程 第 1 讲 编程范式

本讲座的英文版内容基于教材：

Y. Chen, G. De Luca, 《面向服务的计算和系统集成：软件、物联网、大数据喝人工智能服务》(Service-Oriented Computing and System Integration: Software, IoT, Big Data, and AI as Services), 第 8 版, Kendall Hunt Publishing, 2022 年。

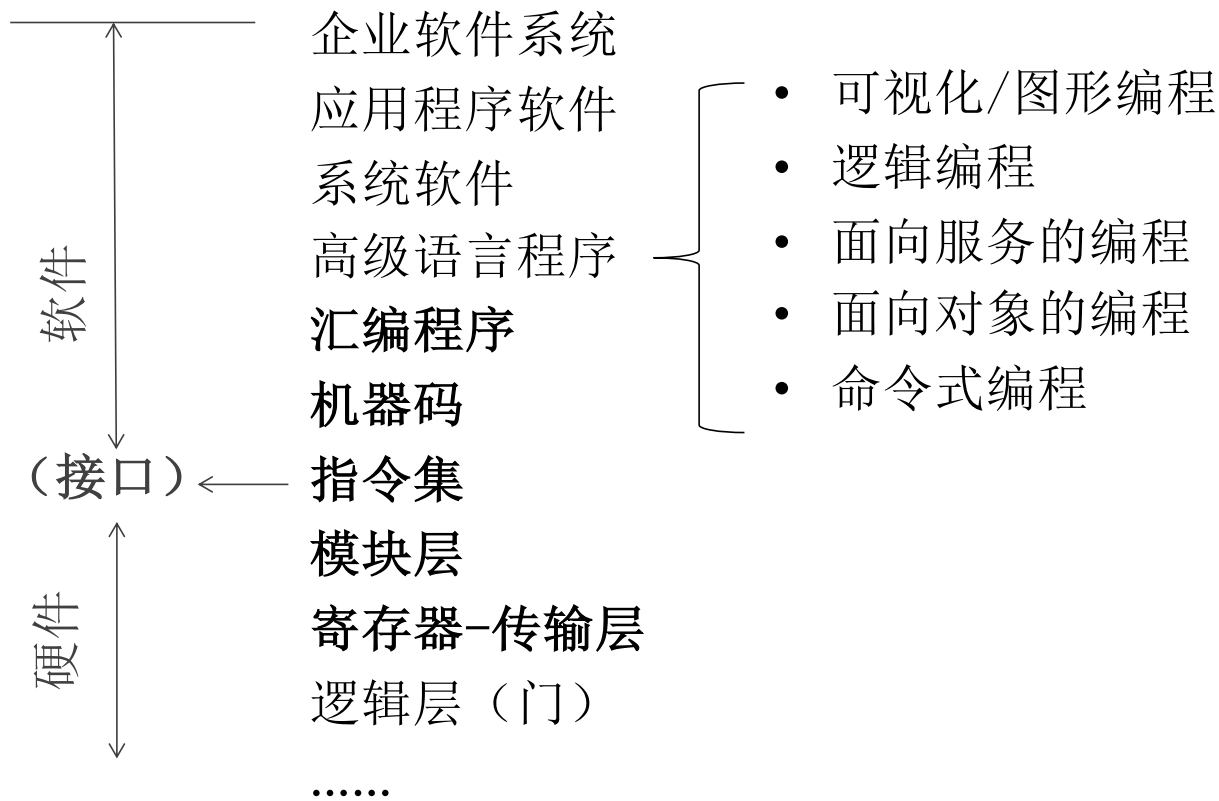
<https://www.public.asu.edu/~ychen10/book/socsi.html>

本讲大纲

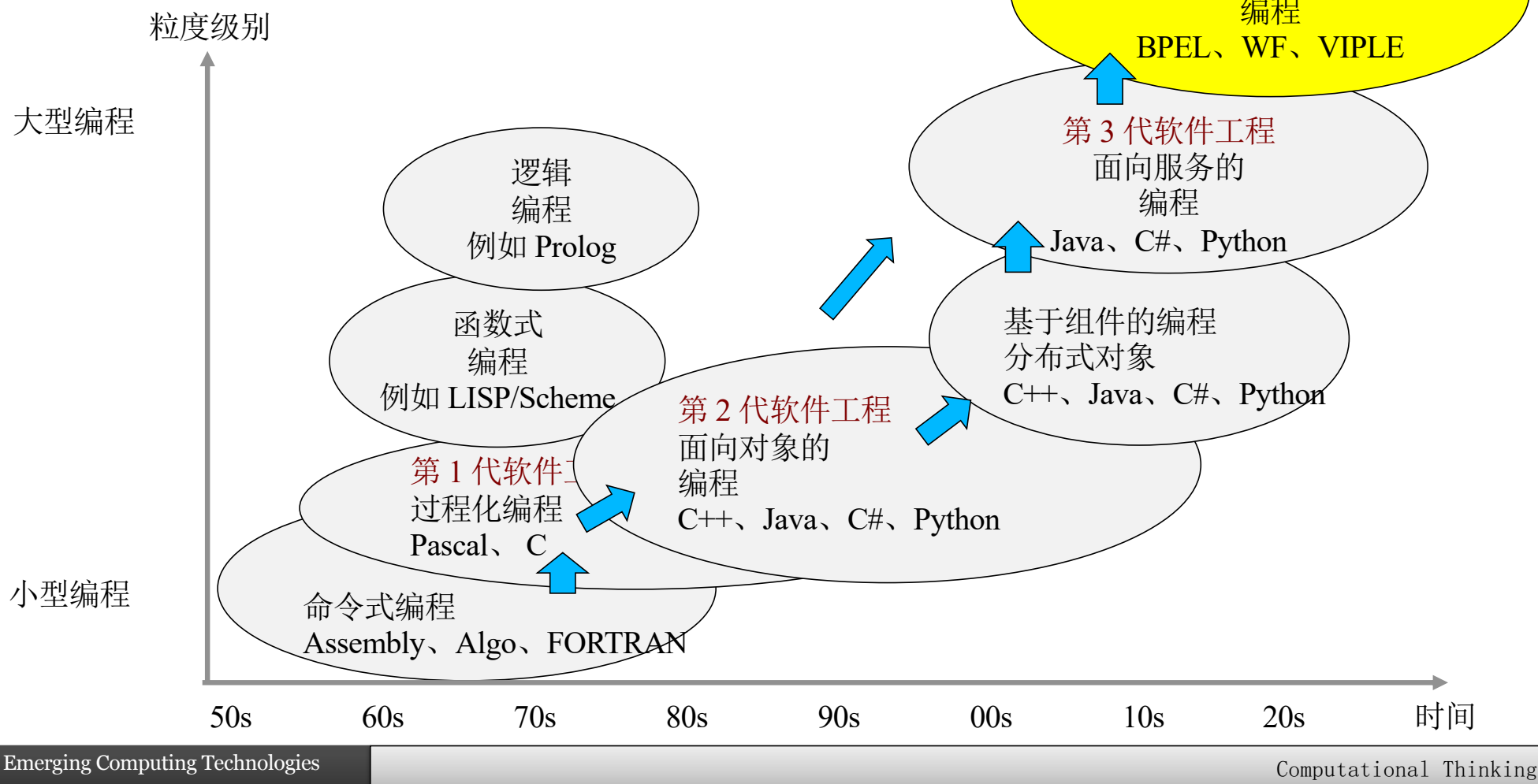
学习

- ❑ 编程语言和编程范式
- ❑ 工作流和有限状态机
- ❑ 亚马逊简单工作流 SWF
- ❑ 可视化编程语言在教育中的应用

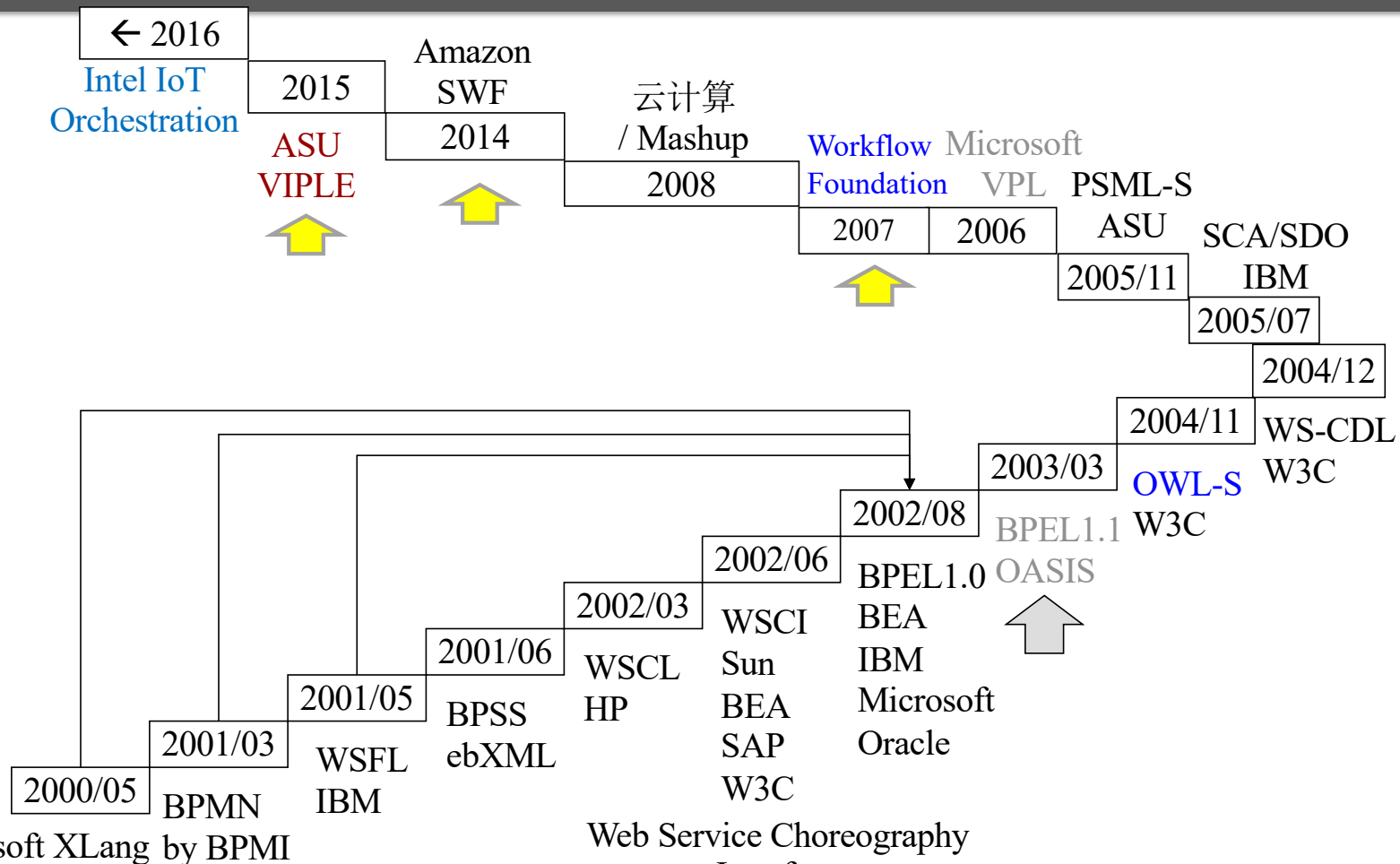
编程语言的层次和编程范式



编程范式

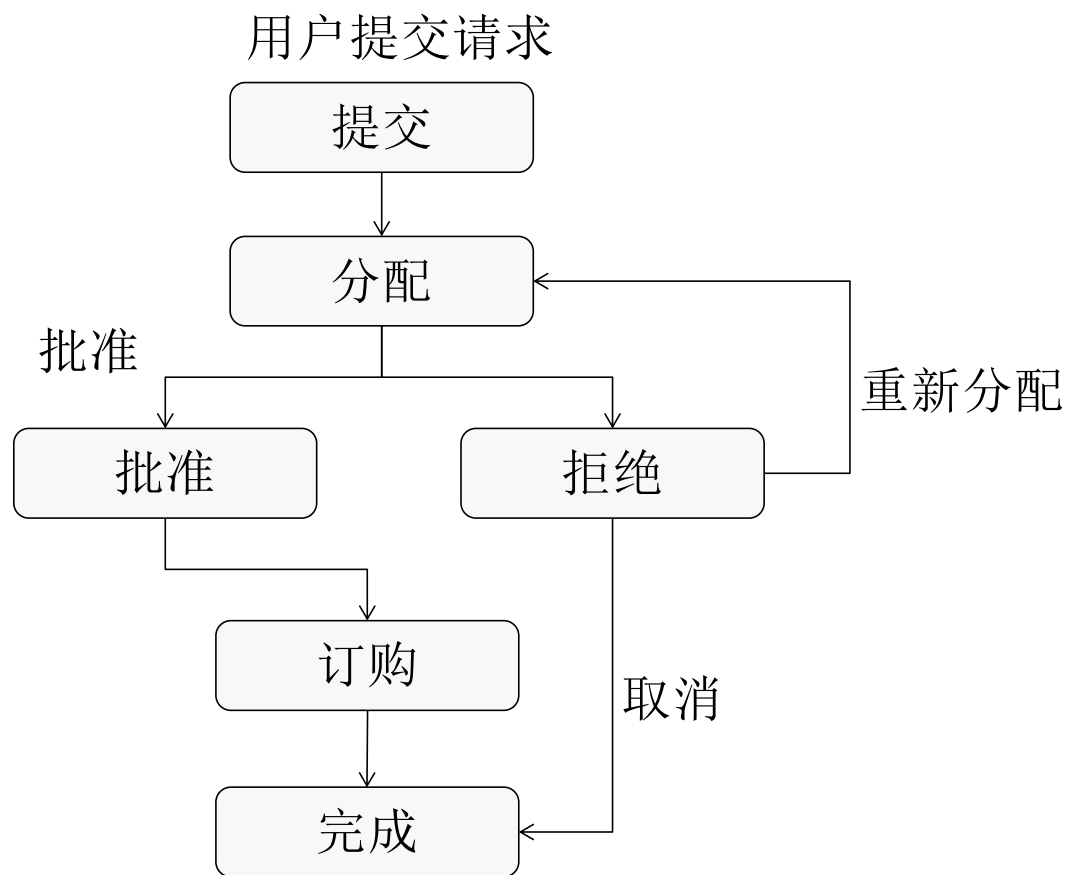


工作流/可视化/图形语言

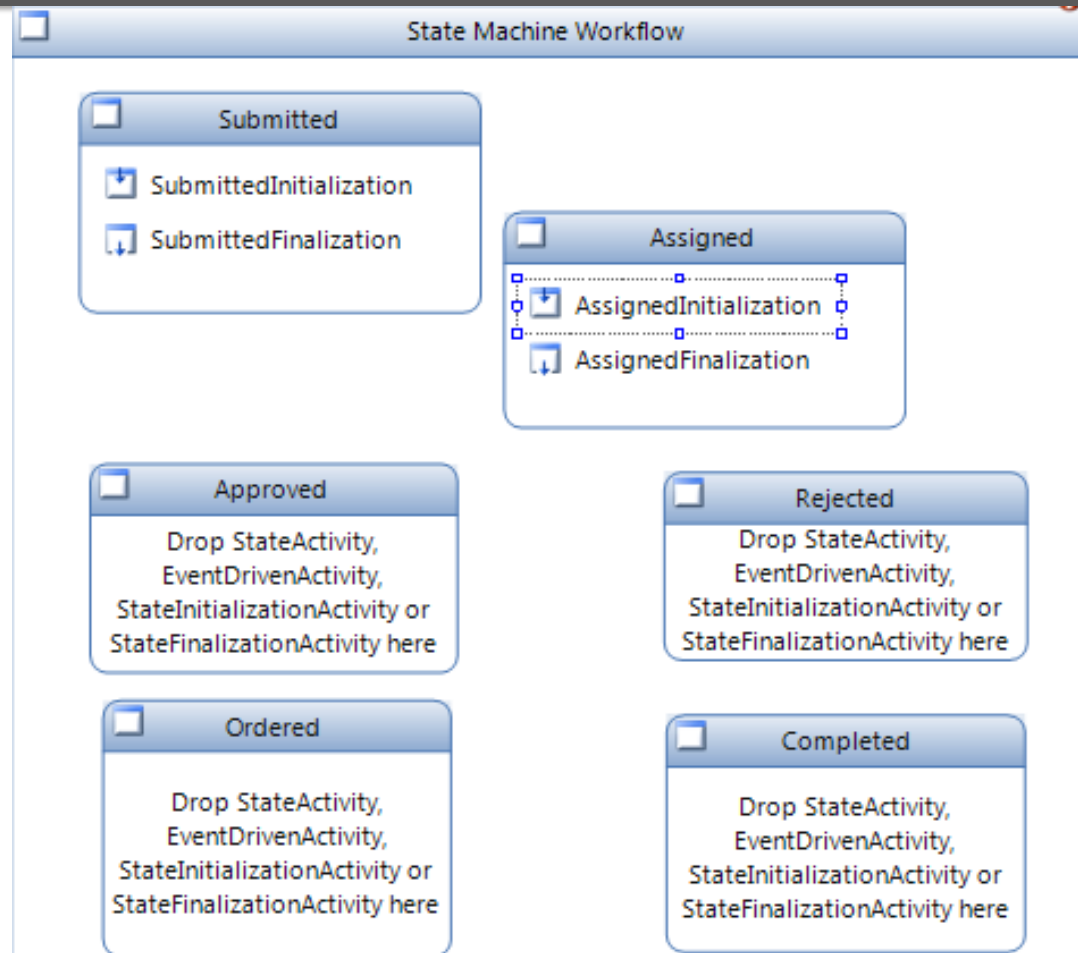
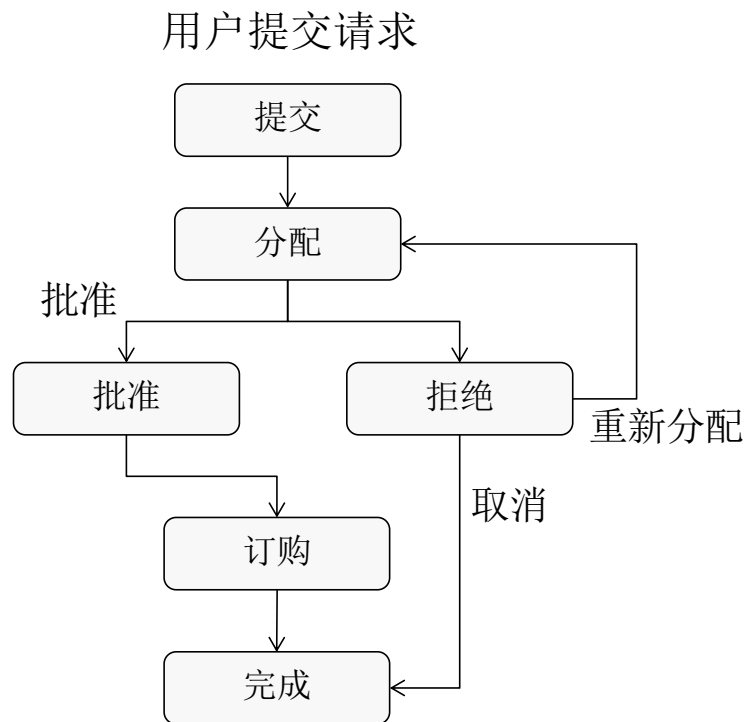


一家简单的网上商城的流程图

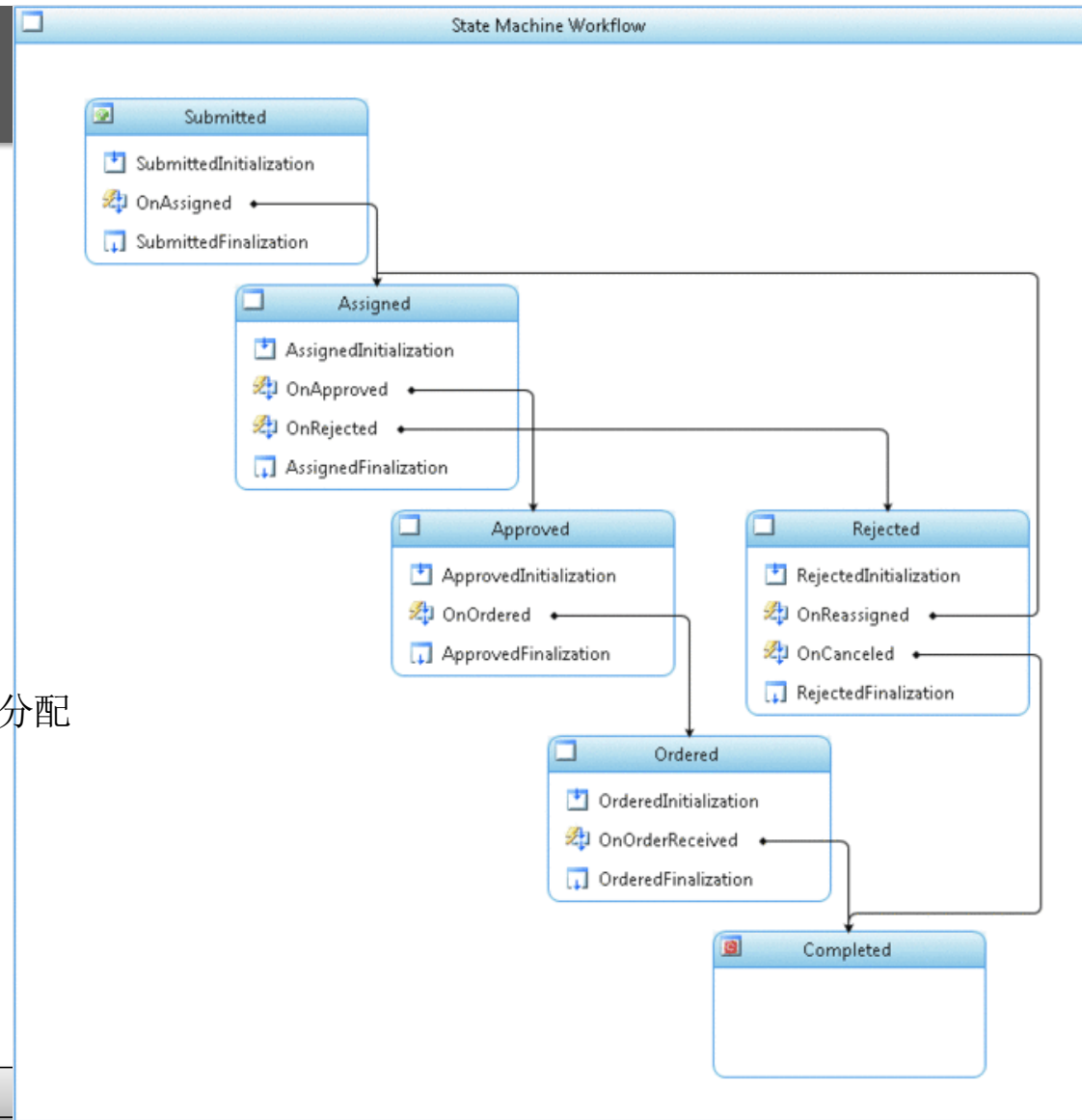
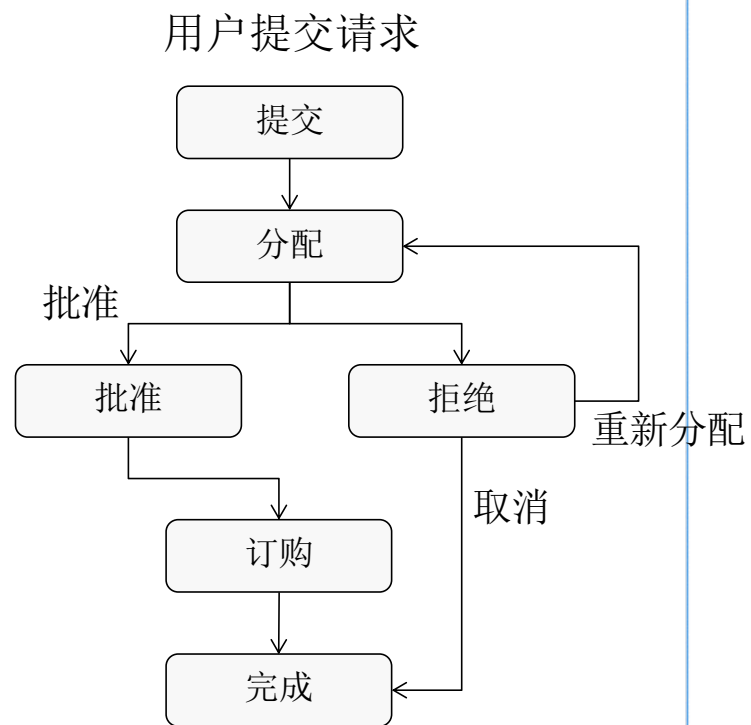
在线订购流程



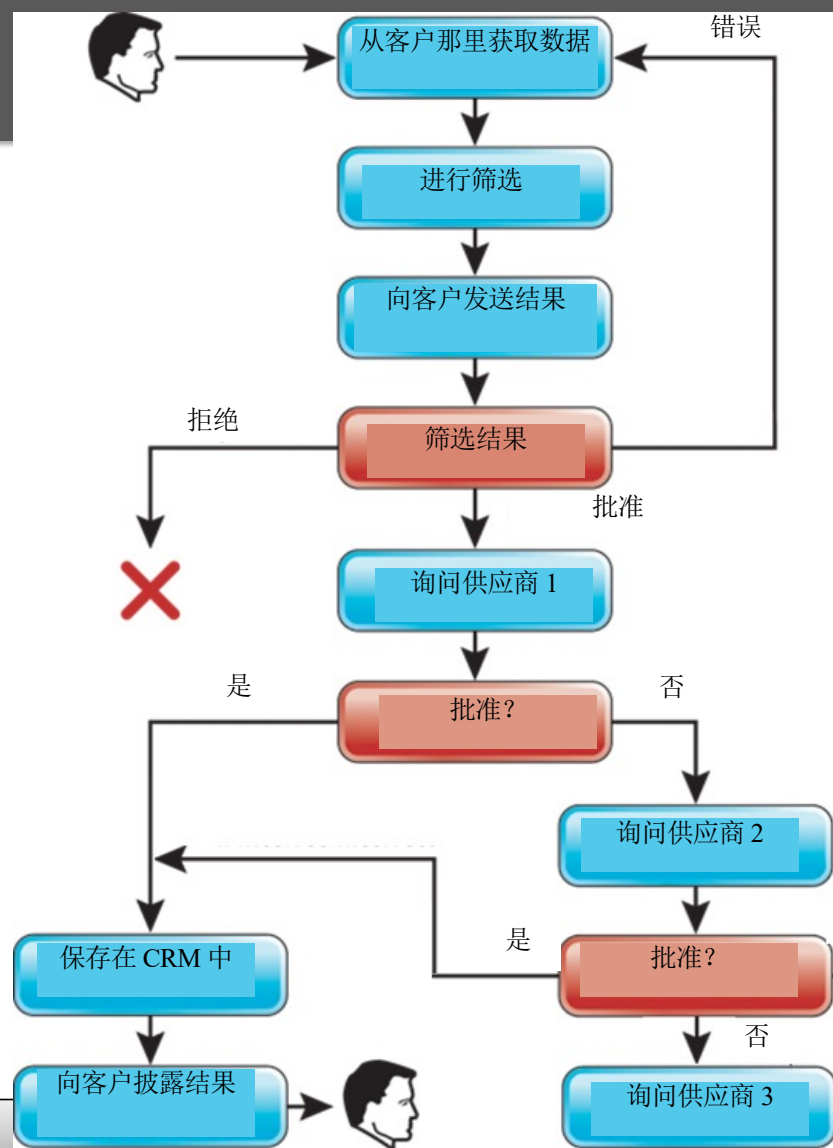
定义有限状态机的状态



定义状态之间的转换

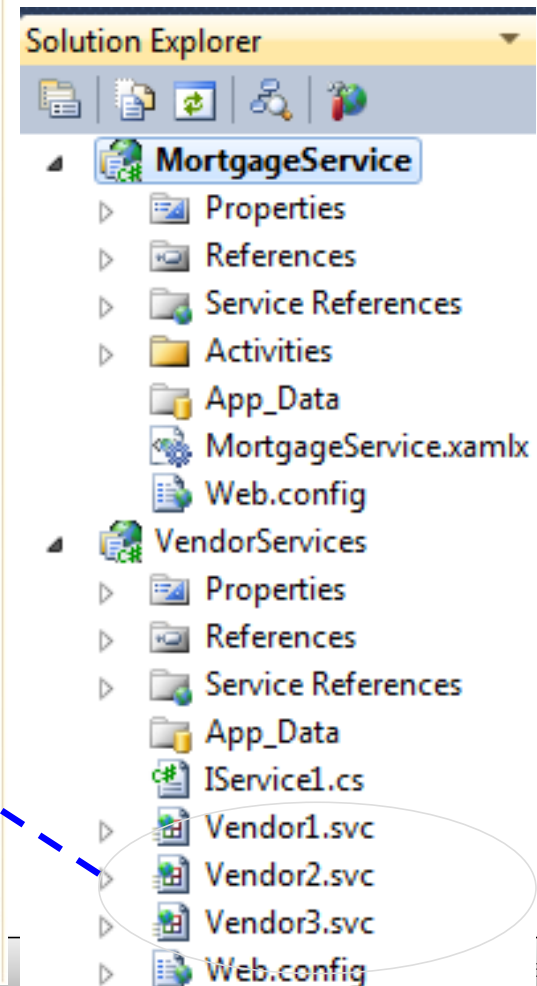
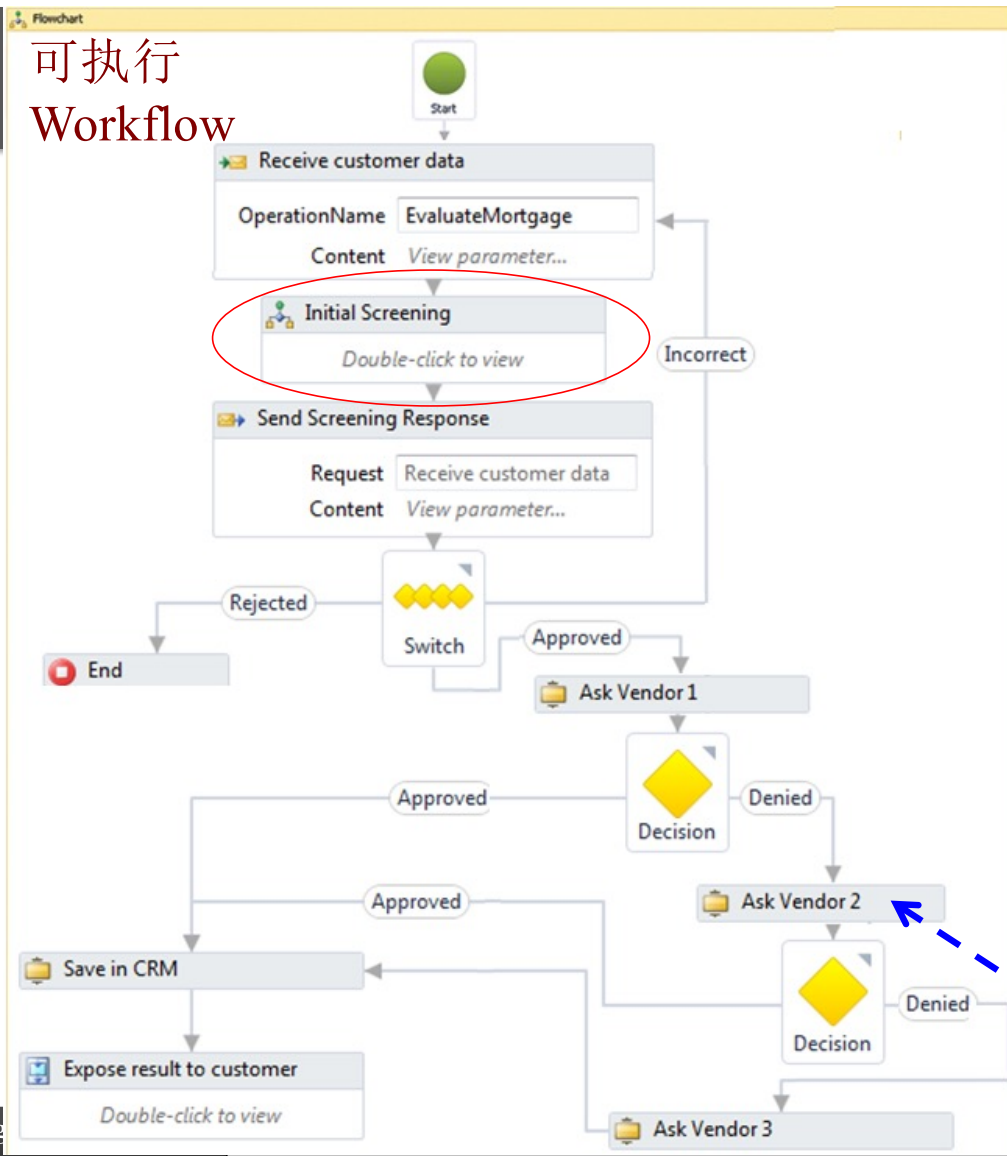


按揭申请网站的流程图

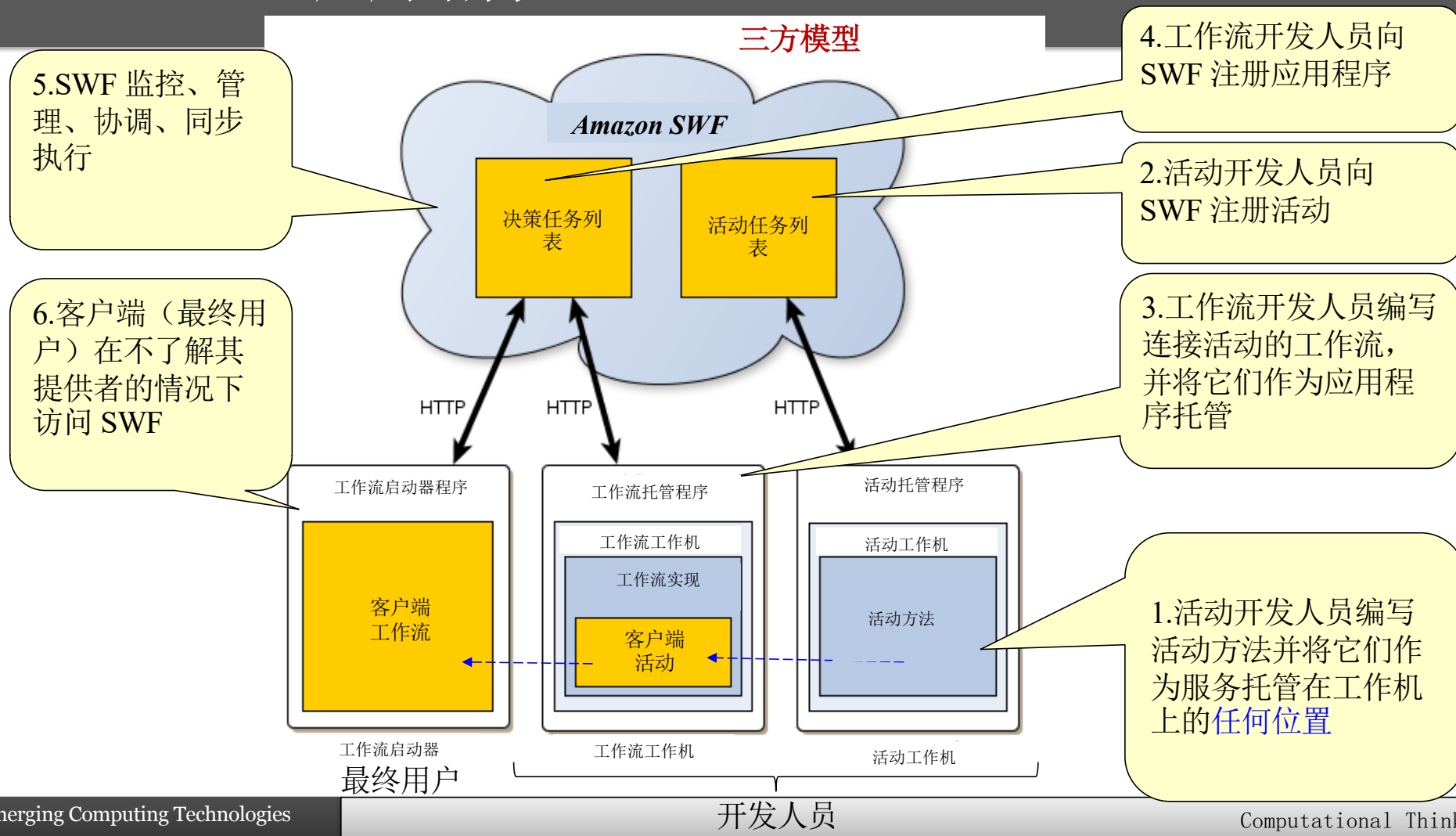


资料来源:
<http://msdn.microsoft.com/en-us/magazine/ff646977.aspx>

可执行 Workflow



Amazon SWF 应用结构



在 Amazon SWF 中创建你的工作流

1. 编写工作流中所需的活动方法或找到现有的。托管你自己的服务 **Service Provider**
 2. 编写一个决策器来实现工作流的协调逻辑。 **Service Consumer**
 3. 使用 Amazon SWF 注册你的活动和工作流。你可以通过编程方式或使用 AWS Management Console 来执行此步骤。 **Service Broker**
 4. 启动你的活动工作机和决策器。
 5. 开始你的工作流的一项或多项执行。
 6. 使用 AWS Management Console 查看工作流的执行情况。
- 注意：上述流程表明你同时开发了活动和工作流。通常情况下，你不会同时开发这两项。

workflows/可视化编程语言在教育中的应用

- 今天的大多数 workflow 语言都是可视化的；
- 教育行业则会使用简化的 workflow 语言：



MIT: Scratch - 可视化游戏编程



Carnegie Mellon: Alice - 可视化游戏编程



MIT/Google App Inventor: 可视化手机App编程



Lego NXT & EV3 - 可视化机器人编程



Intel IoT SOL: 可视化IoT编程



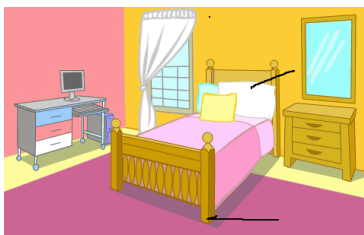
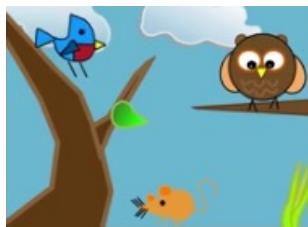
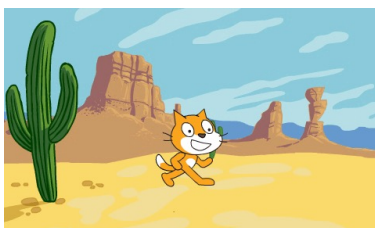
ASU VIPLE: 可视化IoT/机器人编程环境



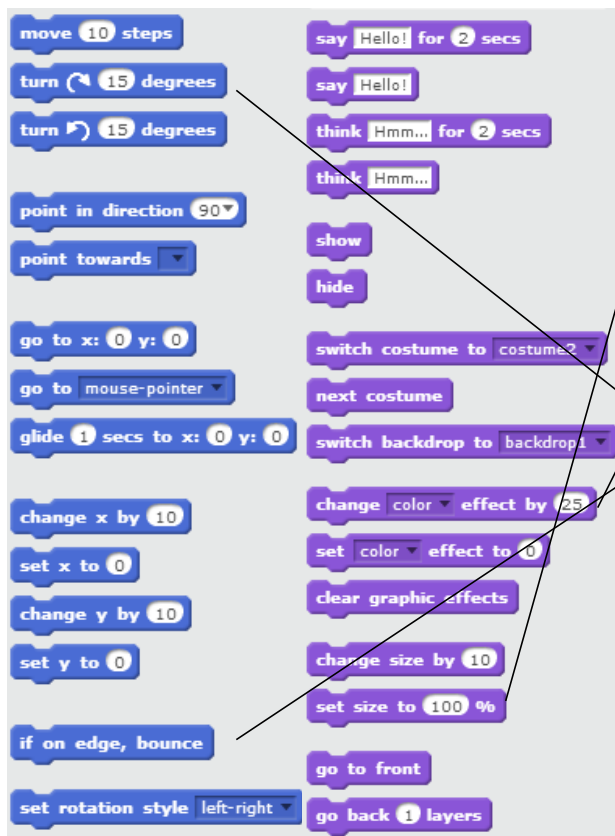
MIT: Scratch - 可视化游戏编程环境

<https://scratch.mit.edu/>

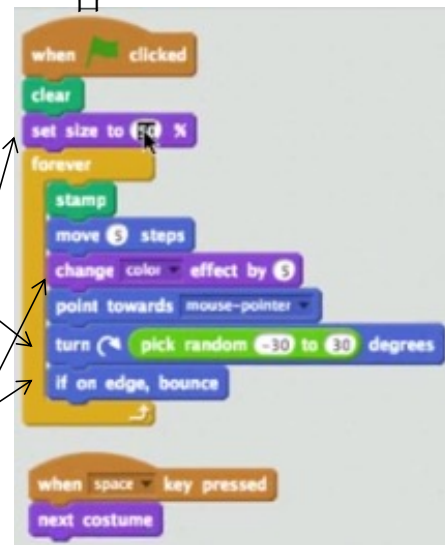
1. 选择人物与场景



2. 选择人物的功能



3. 定义功能的顺序和组合



■ 适合小学生编程



Alice - 可视化游戏编程环境 <http://alice.org/>

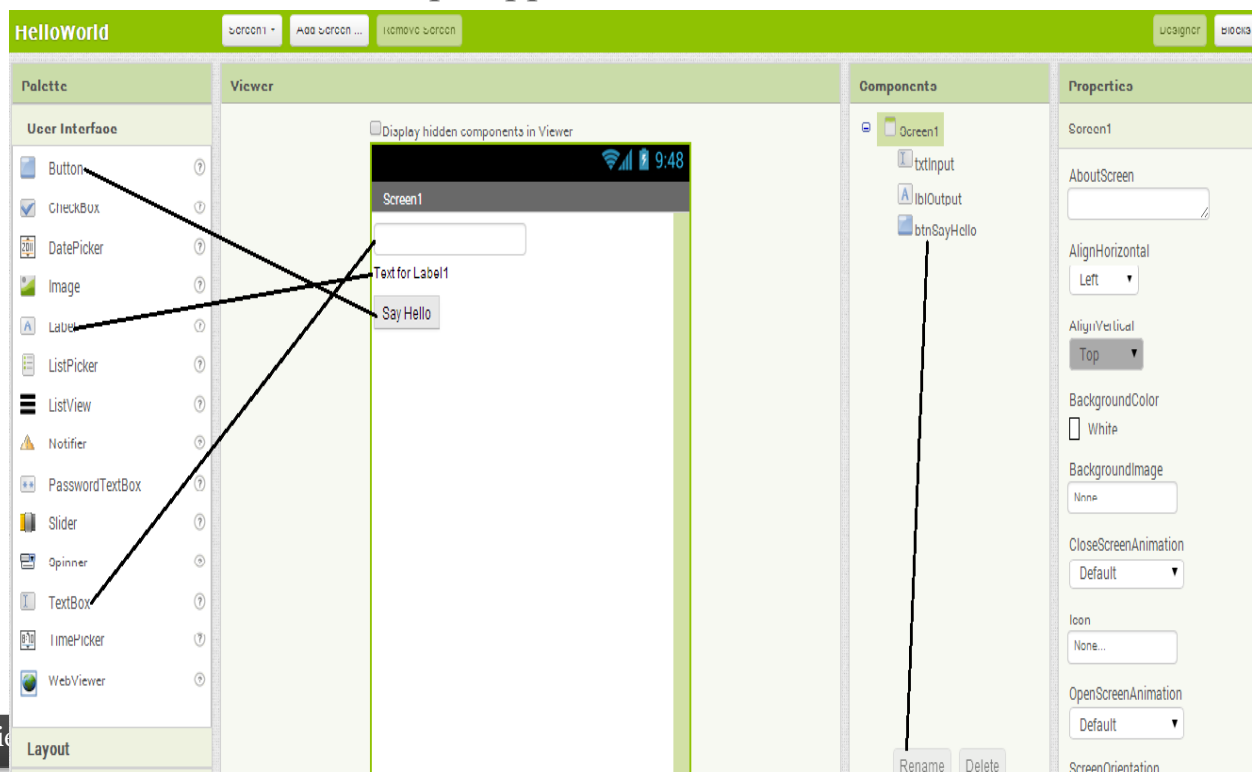
The screenshot displays the Alice 3.1 interface. On the left, a 3D scene window shows three characters: a Mad Hatter, a March Hare, and a Cheshire Cat. Below this is a 'Procedures' panel with a search bar and a list of actions for 'this.marchHare', including 'say', 'think', 'move', 'moveToward', 'moveAwayFrom', 'moveTo', and 'place'. The main workspace on the right is titled 'Scene' and contains a 'myFirstMethod' procedure. This procedure is defined with a 'do in order' block containing a 'turn RIGHT' action for 'this.madHatter' and a 'move' action for 'this.marchHare'. A dropdown menu for the 'move' action is open, showing options for 'direction' (LEFT, RIGHT, UP, DOWN, FORWARD, BACKWARD) and 'amount' (0.25, 0.5, 1.0, 2.0, 10.0). Below the main workspace, there are additional 'do in order' blocks for 'do together' and 'do in order' containing further actions for the characters.

1. 选择场景
2. 选择人物/演员
3. 定义演员的动作
4. 定义时序与协同



MIT/Google App Inventor 可视化手机App编程

- ❑ 基于 Web 的图形用户界面设计
- ❑ 使用拖放操作的可视化编程
- ❑ 模拟器或实体安卓手机
- ❑ 网站: <http://appinventor.mit.edu>





App GUI 背后的代码

Built-in

Control

Logic

Math

Text

Lists

Colors

Variables

Procedures

Screen1

TableArrangement1

txtStock1

txtStock2

lblValue1

lblValue2

lblValue1past

lblValue2past

btnGetStockQuote

Web1

initialize global Past1 to " 0 "

initialize global Past2 to " 0 "

when btnGetStockQuote .Click

do

set Web1 . Url to join " http://finance.yahoo.com/d/quotes.csv?f=l1&s= "

txtStock1 . Text

call Web1 .Get

set Web2 . Url to join " http://finance.yahoo.com/d/quotes.csv?f=l1&s= "

txtStock2 . Text

call Web2 .Get

set lblValue1past . Text to get global Past1

set lblValue2past . Text to get global Past2

Returns the value of this variable.

when Web1 .GotText

url responseCode responseType responseContent

do

if

get responseCode = 200

then

set lblValue1 . Text to join " Current Value: "

get responseContent

set global Past1 to join " Past Value: "

get responseContent

else

set lblValue1 . Text to " Error "

调用 RESTful Service

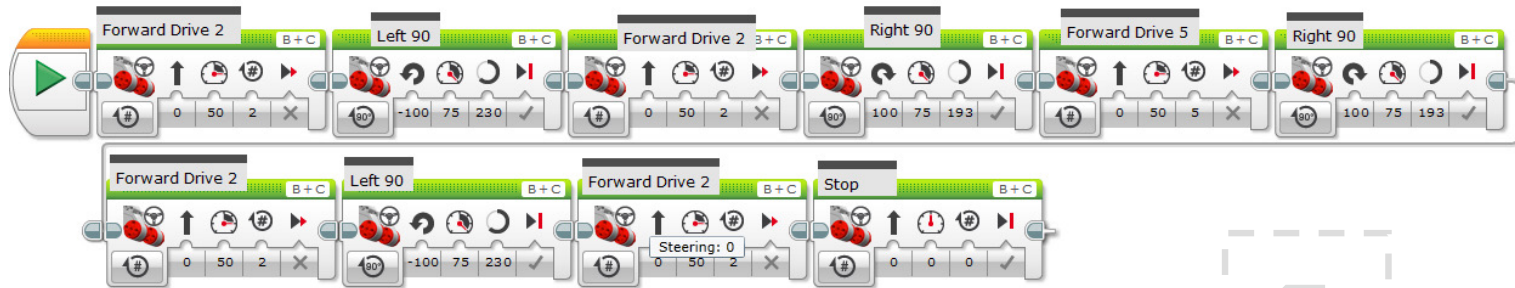
处理 Web 服务的返回值

Show Warnings



Lego EV3 – 可视化机器人编程

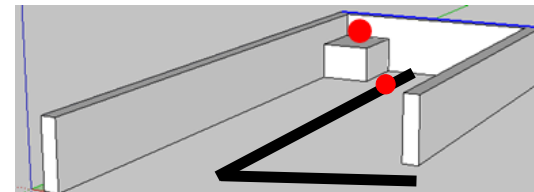
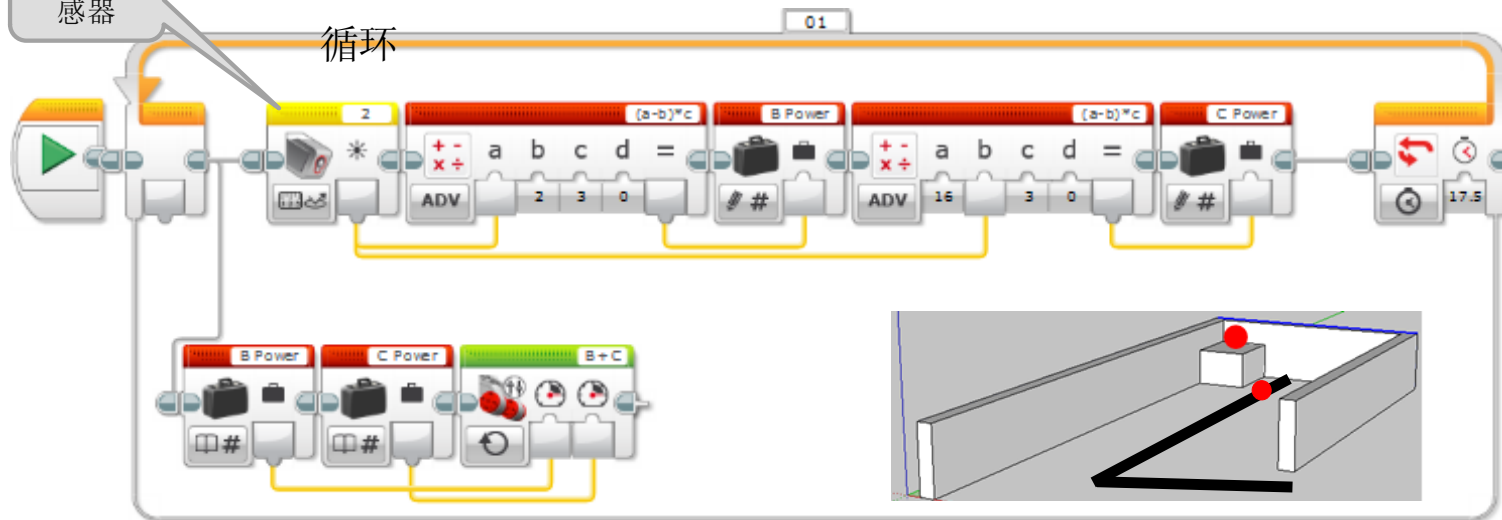
EV3 避障程序



颜色传感器

循环

使用颜色传感器的巡线程序

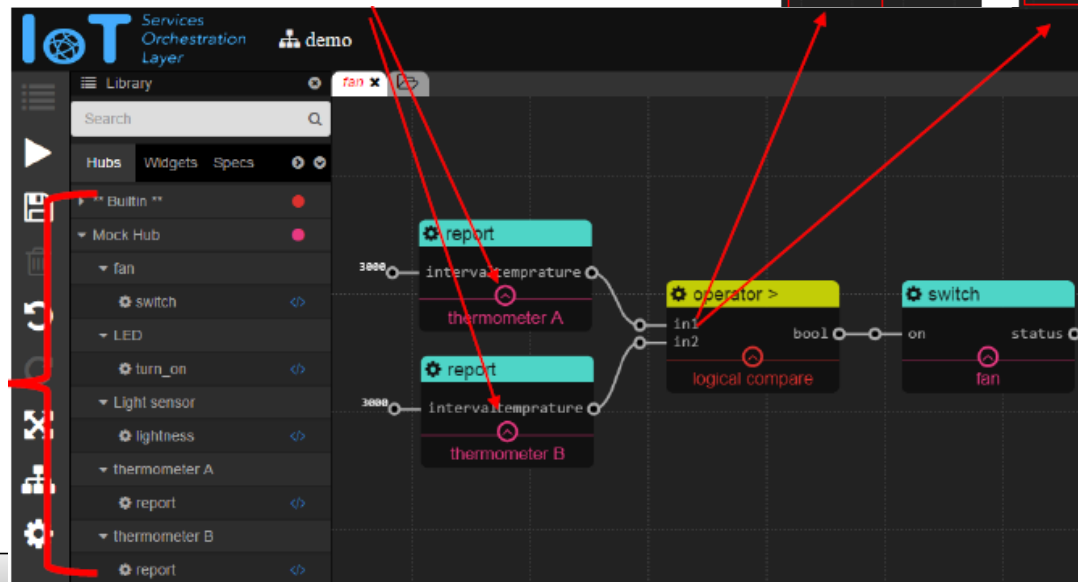
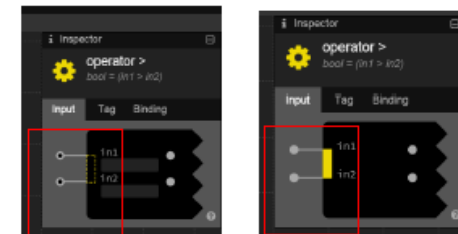




Intel IoT Service Orchestration Layer 物联网编程环境

<https://github.com/intel/intel-iot-services-orchestration-layer-dev>

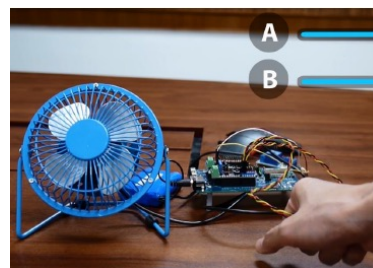
- ❑ 提供强大、可扩展、高质量的解决方案以快速创建物联网应用程序的可视化/工作流方法
- ❑ 与英特尔 Edison 物联网板无缝配合





Intel IoT Service Orchestration Layer

□ 演示 - 添加用户界面和 workflows



设计用户界面

HTML5 编写的工作流程

ASU VIPLE 支持仿真和实体机器人平台

自动驾驶



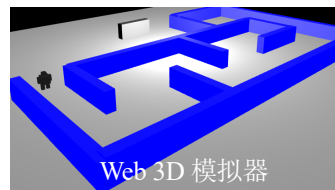
交通模拟



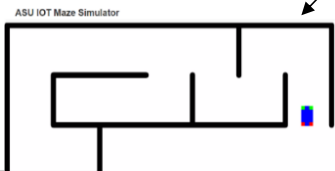
Unity
模拟器



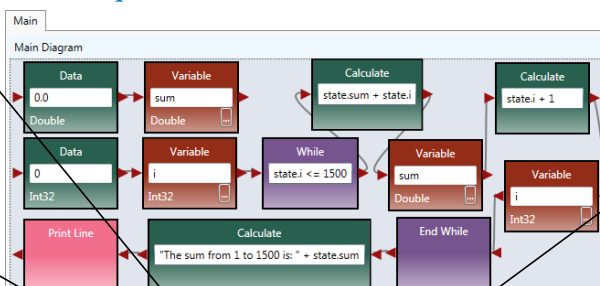
Web3D
模拟器



Web2D
模拟器



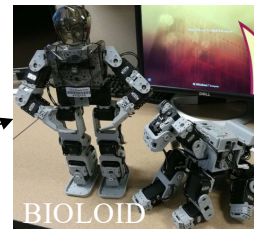
<http://venus.sod.asu.edu/VIPLE/>



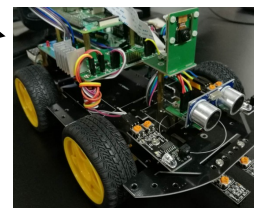
飞行数据训练
与识别



导盲犬和人
形机器人



智能汽车



Lego
EV3



Emerging Computing Technologies Web 2D 模拟器

Intel Galileo 机器人

Intel Edison
机器人

pcDuino 机器人

Computational Thinking