



Adobe Flex 4全新概览

SDK, Spark布局, FXG格式, 开发工具



ADOBE® FLASH® PLATFORM

Zerlot Ma

Flash平台技术经理

议程

- Flash Platform 概要
- Flex 4 SDK 及 Framework
- Flex 4 Spark布局
- 未来的FXG格式
- Flash Catalyst和Flash Builder 4
- 移动设备上的Flex框架 – Slider
- Q & A

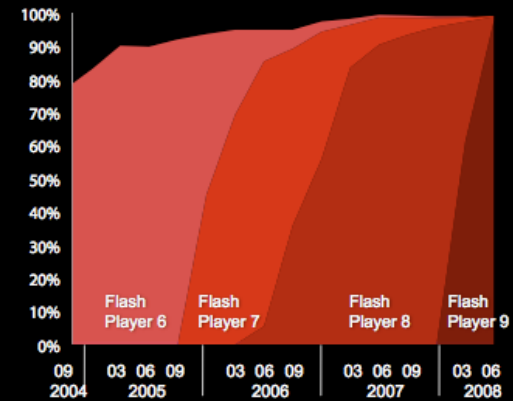
The logo consists of the letters 'Fx' in a bold, white, sans-serif font, centered within a dark gray square with a subtle gradient.The logo consists of the letters 'FB' in a bold, black, sans-serif font, centered within a medium gray square with a subtle gradient.The logo consists of the letters 'FC' in a bold, black, sans-serif font, centered within a light gray square with a subtle gradient.

Flash Platform 简要介绍

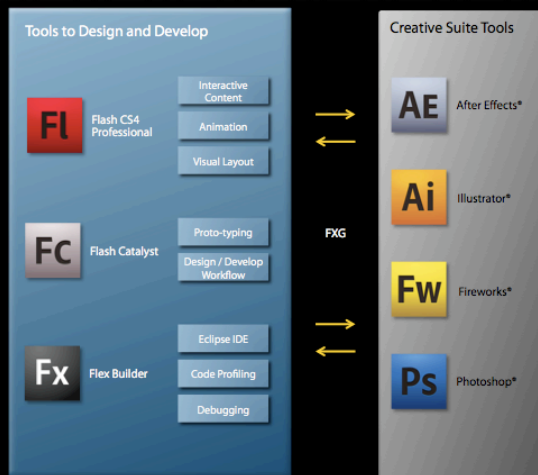
完整的系统



保持一致的传播方式



灵敏的设计与开发协作流程



配套的技术



Adobe Flash Platform

Applications, Content and Video



Tools to Design and Develop



Adobe® Flash® CS4
Professional



Adobe® Flash®
Catalyst™



Adobe® Flash®
Builder™

Framework



Adobe® Flex®

Clients



Adobe® AIR®



Adobe® Flash® Player

Services



Adobe Flash
Platform Services

Servers



Adobe® Flash® Media
Server Family



Adobe® LiveCycle®
Data Services



Adobe Flash Platform Services

分发

分发，提升，跟踪和监视应用，通过社会交往网络，移动社会和桌面来统一完成

协作

增强应用之间的实时协作能力，包括聊天，音频和视频

社会化

允许应用与领先的互联网社交应用整合，如 Facebook, Twitter 等类型的应用

“Shibuya”

监控应用的支付模式和一系列的业务发展

Flash Platform 工具定位

实现开发者和设计师在创作交互内容和应用时高度协作



面向设计人员和交互开发人员的创作工具，创建高度自由的，引人入胜的交互内容

创建 引人入胜的体验和视频
媒体内容



面向设计师的交互设计工具，无需额外编码即可快速创建Flex应用的用户交互界面

转换 设计方案到功能性
交互应用界面



针对软件开发人员使用Flex框架和ActionScript构造跨平台的网络应用和交互内容的整合开发环境

开发 跨平台的RIA应用

Adobe Flash Platform 工具集合



Flex 4 框架

开源框架，构造Flex应用基础



Adobe Flash Catalyst

专业的交互设计工具，快速创建UI和交互内容



Flash Builder 4

基于Eclipse的高效开发工具，能够构造以数据展示逻辑为中心的Flex应用



Flex 4 SDK及 Framework

The Flex logo, featuring the letters 'F' and 'x' in a bold, white, sans-serif font. The 'F' is significantly larger than the 'x'. The logo is centered within a dark gray square that has a subtle, concentric circular gradient pattern.

Flex 推动力

全球企业

金融

消费类点子

医疗保健

银行

出版

前10名中的8个

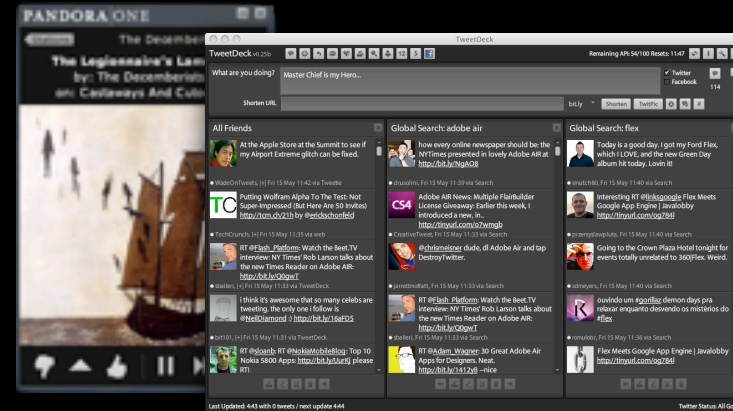
前10名中的8个

前10名中的8个

前10名中的7个

前10名中的7个

创业公司



独立软件供应商

前10名中的8个

开发者!

150% 的年增长率

Flex Momentum





强调设计

- Spark 组件模型
- MXML 2009
- FXG



提升开发者生产力

- 编译器性能
- 双向绑定
- CSS 增强

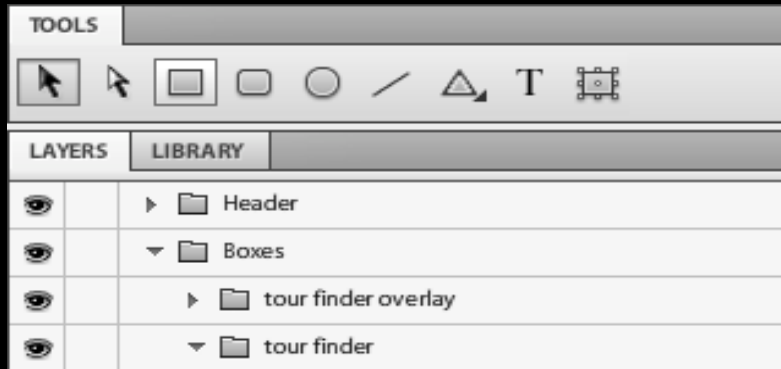


框架进化

- 增强状态，特效和布局
- 高保真字体框架
- 新的视频组件

强调设计?

- 一个设计师接触Flex时想能够找到下面的编辑方式:



...但是却看到下面的代码

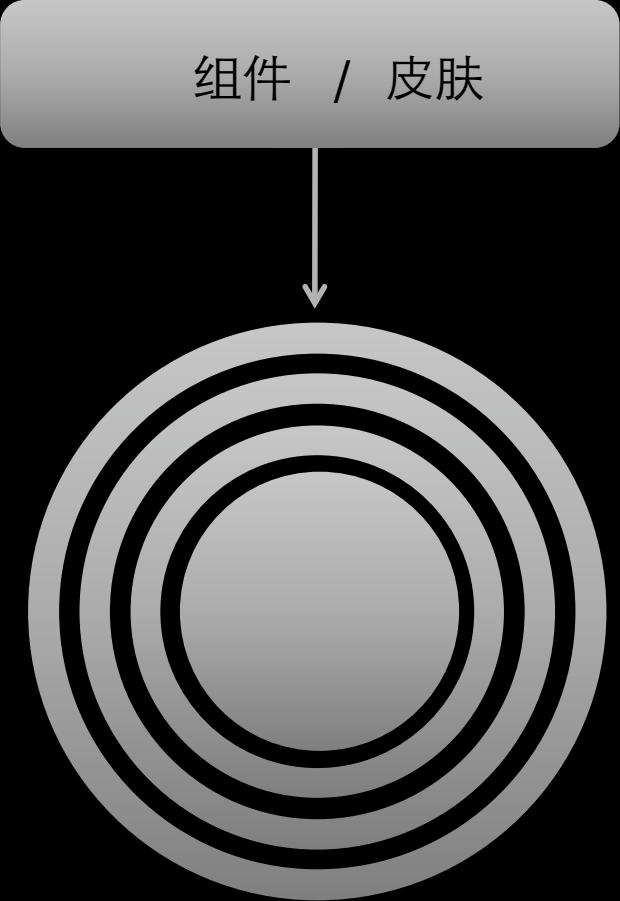
```
<Canvas>
<Label text="name:" />
<Button />
<DataGrid />
</Canvas>
```

- 设计师需要自由创意
- MXML需要以下支持:
- 原始图形
- 灵活的, 出色的布局
- 丰富的动画特效和状态

Flex UI Framework

Flash / AIR runtime

Halo 组件模型

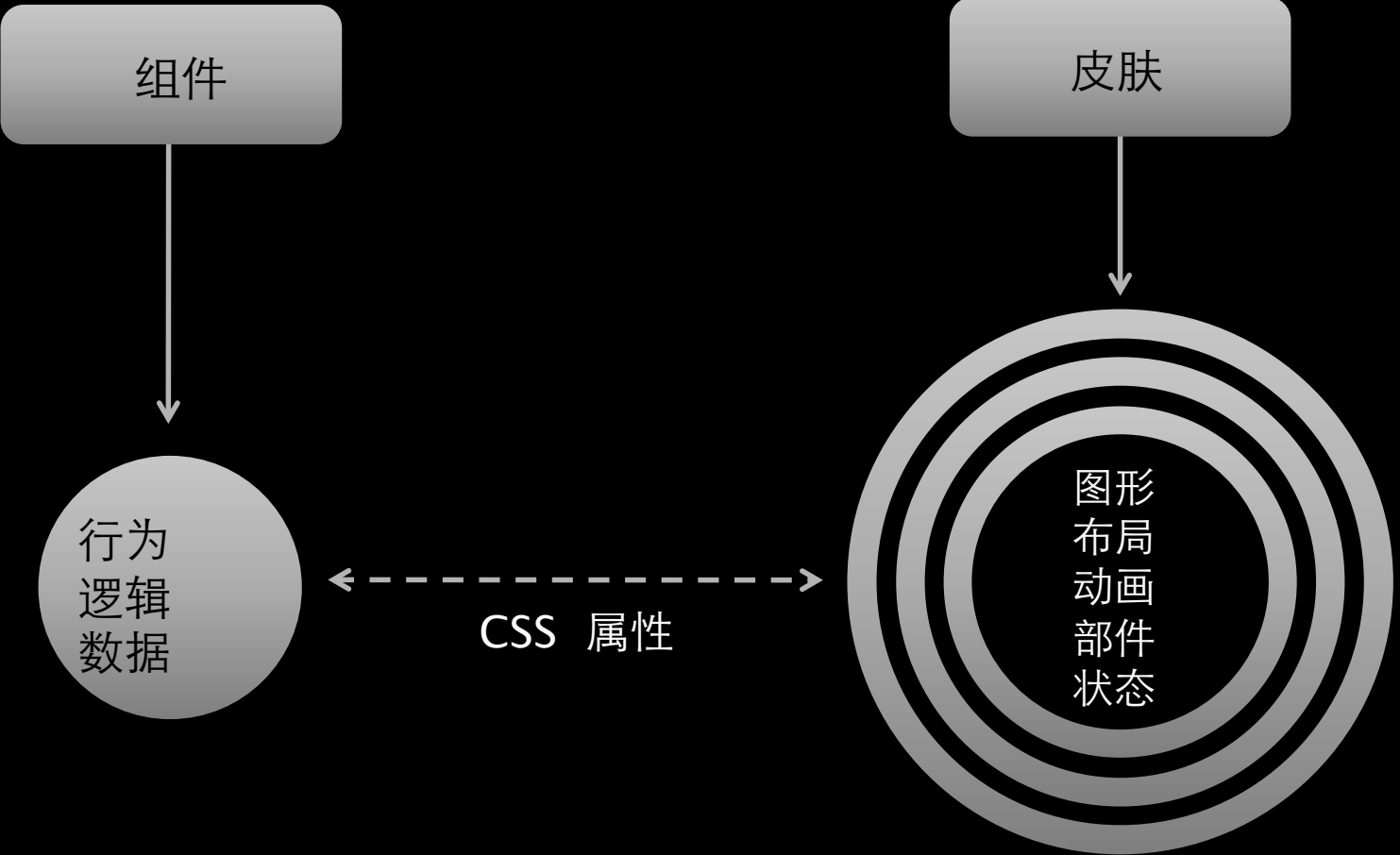


图形
布局
动画
部件
状态
行为
逻辑
数据

Spark 组件模型

ActionScript

MXML



从外观中剥离行为逻辑

- 针对重复组件，重复标签的解决方案
- 生产力提高
- 修正模棱两可的环节
- **MXML 2009**
 - MXML语言新功能只存在于2009定义中
 - 每个文件一个语言命名空间，但是可以混合和匹配到一个应用
 - 在2009定义中，Spark和Halo并存
 - Spark 只能在2009定义中工作- 不能在2006定义中工作
- `xmlns:fx="http://ns.adobe.com/mxml/2009"`

2009定义中的改变: 语言vs. 组件

- `xmlns:fx="http://ns.adobe.com/mxml/2009"`
 - 语言标签 (Script, Style, 其他)
 - 内置类型 (Array, String, Number等)
- `xmlns:s="library://ns.adobe.com/flex/spark"`
 - 所有在Flex4中引入的新类
 - RPC
 - 图形
- `xmlns:mx="library://ns.adobe.com/flex/halo"`
 - 所有Halo遗留组件
 - 状态, 验证类, RPC等
 - 任何2006定义中的东西
- `xmlns:mx="http://www.adobe.com/mxml/2006"`
 - 跟Flex3的定义一样

- 双向绑定

- `text="@{myModel.data}"`

- `<fx:Binding source="foo.text" destination="myModel.data" twoWay="true" />`

- 高级 CSS

- 多个类选择器

- ```
<Button id="upButton" styleName="default tiny" />
```

- 类型选择器

- ```
#upButton { fontSize: 14 }
```

- 子选择器

- ```
s|Scrollbar #upButton { baseColor: #FF8888 }
```

- 伪类型选择器

- ```
s|Scrollbar #upButton:over { baseColor: #8888FF }my
```

- 难以使用
 - 代码冗长
 - 层次结构难于管理，分解和加工
 - 很难做优化

```
<states>
<State name="login">
<SetProperty target="{goBtn}" name="label"
value="..." />
</State>
<State name="register">
<SetProperty target="{goBtn}" name="label"
value="..." />
<AddChild target="grp">
<Checkbox label="Agree to terms" />
</AddChild>
</State>
</states>
```

```
<Group id="grp" >
<TextBox text="username:" />
<TextInput />
<TextBox text="password:" />
<TextInput />
<Button label="new user?" />
<Button id="goBtn"
</Group>
```

- 用State标签声明状态
- 在语言中描述“可交替的视图”
- 更改值，绑定和事件侦听
- 引入和排除组件，尽可能容易的设定不同状态下的可见性
- 不用为所有状态定义完全的'default'状态
- 在2009定义中使用; 2006定义仍然支持传统states

```
<states>
<State name="login"/>
<State name="register"/>
</states>

<Group>
<TextBox text="username:" />
<TextInput />
<TextBox text="password:" />
<TextInput />
<Button label="new user?" />
<Checkbox includeIn="register"
label="agree to terms" />
<Button label="log in"
label.register="sign up" />
</Group>
```

- 特定的目标动画 (不仅仅是UIComponents)
- 具备自定义内插值 (内插曲线, 抛物线等) 内插特定属性的动画
- 自动反转特效
- 支持Pixel Bender滤镜
- 3向平面内变形
- 精确复杂的Timing模型, 支持关键帧

- FTE:全新底层字体渲染引擎（基于FP10）
- TLF: 全新字体布局类库（基于FTE）
- 优势:
 - 软连字符(不可见连字符，比如word里的换行和间隔符)
 - 精确基线控制 (e.g., superscripts和subscripts)
 - 居右，居中和小数点制表符
 - 垂直判定
 - 多列文字排列
 - 连字，大写样式，数字样式
 - 整合渲染设备字体 (能阴影或渐变我的设备字体)
 - 双向字体，横向和纵向



ascender line
cap line

midline

baseline

descender line

ascender line
cap line

midline

baseline

descender line



Flex 4 Spark布局

The Flex logo, featuring the letters 'F' and 'X' in a bold, white, sans-serif font, set against a dark gray, circular gradient background.

What is Spark?

- Flex4 中的全新组件和皮肤架构
- 新命名空间
 - xmlns:s= “library://ns.adobe.com/flex/spark”
- 针对旧组件架构下的开发问题而出现
 - Skinning
 - CSS
 - 组件、状态、文字、图像定义
 - 布局定义

Spark Layouts

- Halo Layouts in Flex3
- 继承Halo Layouts的优点
 - 相似的执行流程
 - API
 - 布局逻辑
 - 完全适应Flex 3开发人员的习惯
- 提供系列附加功能和变动
 - 更加适合模块化布局设计

Spark Layout: Common characteristics

- MXML开发工作没有大变动
 - width,height,minWidth,explicitWidth,percentWidth等属性未改变
 - left,right,top,horizontalCenter,baseline样式仍支持
- 开发者可以找到非常熟悉的环节
 - 核心LayoutManager类不变
 - 组件生命周期不变
 - commitProperties(),measure()和updateDisplayList()
 - 仍然被LayoutManager调用
 - 调用顺序不变
 - 验证机制不变
- 组件默认尺寸计算机制方法measure()不变
- updateDisplayList()方法调用
 - 仍然保持更新组件内部子类尺寸

Spark: changes

- Spark Layouts从组件容器中分离出来
 - 发生measure()或updateDisplayList()调用
 - 衡量和安排子元素的任务被指派给Spark布局实例
- 改变表现在
 - 布局逻辑被抽象提取出来，在一个分离的类中
 - 继承LayoutBase类
 - LayoutBase是Spark Layout的最小化布局
 - 引入新API作为Spark Layout和元素衡量尺寸位置的联系
 - 通过ILayoutElement接口实现，支持2D和3D变形
 - 举例：如果你没有创建自定义布局，则无需关心该接口;如果你创建了自定义布局，该接口会帮你节省很多时间
 - 布局可视化，包括创建，销毁，为数据容器回收itemRenderer
 - 全部通过DataGroup容器实现
 - DataGroup，Spark数据容器基础构造元素

Spark improvements and new features

- Spark Layout为了适应
 - 更多的模块
 - 更强的能力
 - 更容易的扩展性
- 可分配的布局
 - 布局逻辑从容器中抽离
 - Spark容器可以被分配以各种不同的Layout
 - 可以在运行时分配
 - 降低容器类的数量
 - 增强模块化和代码重用灵活性
- 举例，在Spark中，只有一个List类，为了获得和Flex 3 Halo Layout中TileList相同的功能，只需要给List类指派一个TileLayout实例即可

Spark improvements and new features

- 自定义布局
 - 从容器中抽离出布局逻辑
 - 新LayoutBase类和LayoutElement接口允许开发者快速创建强大的布局
 - 混合和匹配常用的Spark容器
- 任意的2D变形
 - 常用Spark布局现在支持任意2D变形
 - 内置于ILayoutElement接口
 - 可以通过Spark容器的所有子元素实现

Spark improvements and new features

- 像素级滚动条支持
 - 任何环境下
 - 支持所有DataGroup扩展容器
 - 包括基于Spark布局的自定义布局
- 3D支持
 - LayoutBase类和LayoutElement接口设计上支持
 - 包括网络上几种常用布局3D变形特效
 - CoverFlow (IPOD中常用歌曲虚拟排列书架效果)
 - Carousel (旋转木马效果)
 - WheelLayout, 3维环状布局特效

Spark improvements and new features

- 深度顺序
 - 指定每一个子元素深度
 - 无论是MXML或者是一个自定义布局
- 布局变形
 - 定义x,y,z,rotation,scale，不影响布局效果
 - 所有通用Spark布局都内置2D变形缩放支持
 - 举例，鼠标浮动上一个horizontal list容器内的缩略图，该图形可以轻松实现放大，缩放的变形效果，而整体Layout布局会自动适应这一变化，而不会影响整体布局
- 一致的坐标空间
 - width,height,measuredWidth,measuredHeight总是保持高度协调
 - 变形前后

working with Spark Layouts

- Halo布局容器与Spark布局容器的变动
- HGroup和VGroup是简单Group类，使用HorizontalLayout和VerticalLayout默认值

Halo布局容器	对应Spark布局容器的变更
Canvas	与BasicLayout组合
HBox	与HorizontalLayout组合或者使用HGroup class
VBox	与VerticalLayout组合或者使用VGroup class
Tile	与TileLayout组合
List	使用VerticalLayout的List
TileList	使用TileLayout的List



Flex 4 FXG格式 (Flash XML Graphics)

The letters 'FX' in a bold, white, sans-serif font, centered on a dark gray square background with a subtle radial gradient.

- FXG: 基于MXML的图形格式
- 可以被设计工具解读
- 基于flash渲染模型
- 静态类型, 没有数据绑定, 布局, 事件侦听, 样式等描述
- 编译器优化, 对于提高性能非常重要



- MXML图形库提供丰富的原始支持
 - 简单原始形状
(矩形, 圆角矩形, 椭圆, 圆等等)
 - 复杂的路径
(线性, 二次方程, 贝泽尔曲线)
 - 全范围的填充和描边
(单色, 透明, 位图, 线性和辐射渐变)
 - 蒙版, 滤镜, 混合模式等.
(模糊, 发光, 阴影, 遮掩, 叠加...)
 - 色彩和2维变形
(翻转, 缩放, 色调, 亮度...)
 - 整合的字体, 位图



```
<Graphic>
<Path data=" ..... ">
<fill>
<LinearGradient angle="90"> ....
</LinearGradient>
</fill>
</Path>
<Path blendMode="screen" data="..." >
<fill>
<LinearGradient angle="45"> ...
</LinearGradient>
</fill>
</Path>
<GraphicText text="MXML Graphics">
<filters>
<Glow color="#00FF00" strength="3"
... />
</filters>
</GraphicText>
</Graphic>
```

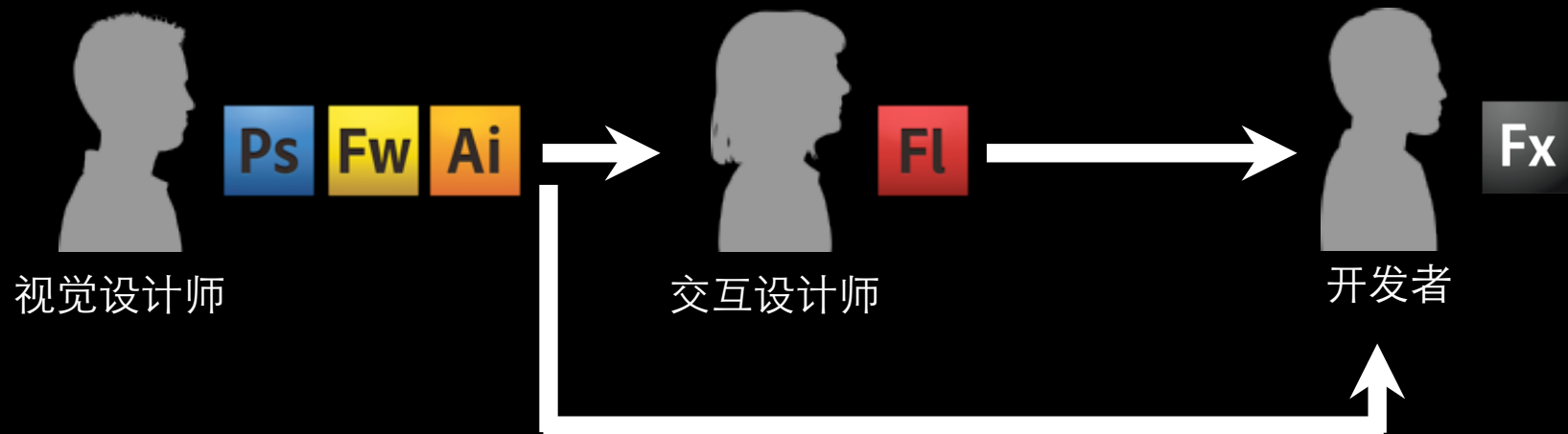


Flash Catalyst

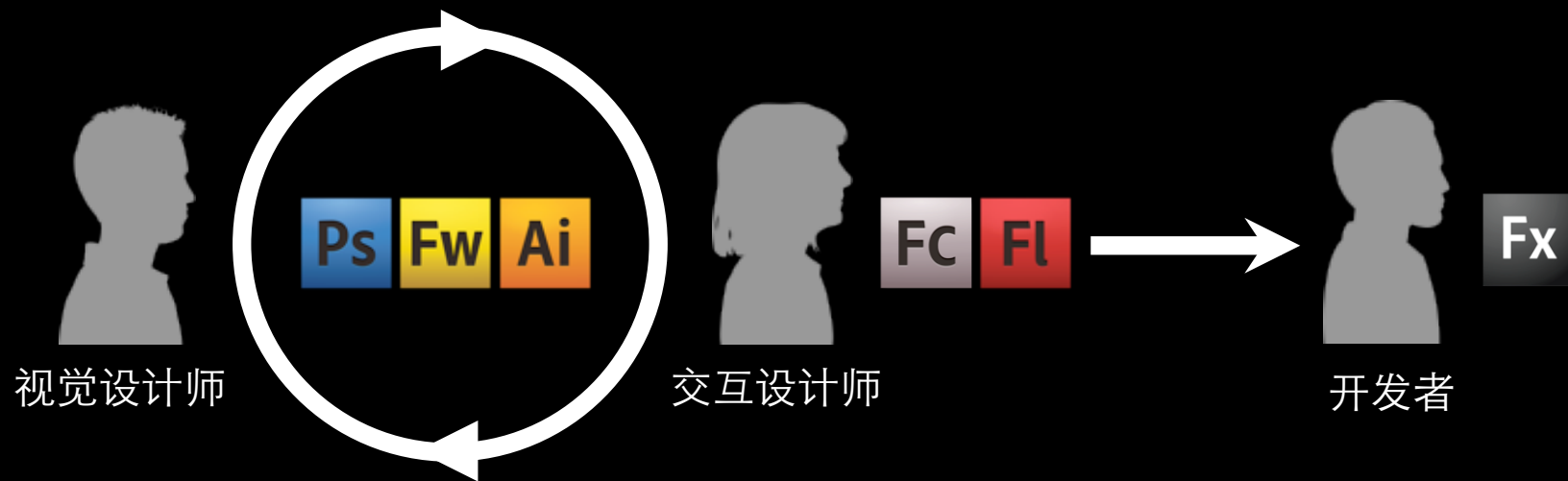
用于快速创建具备用户体验的应用和交互内容的专业设计工具



现在的工作流



新 workflow



workflow 细节



视觉设计师



- 开始使用CS4设计工具进行外观界面设计
- 以高保真的方式将设计稿导入Flash Catalyst
- 将设计元素转换为组件
- 定义状态，交互和动画
- 在图形元素上通过工具间巡回操作来完成提炼

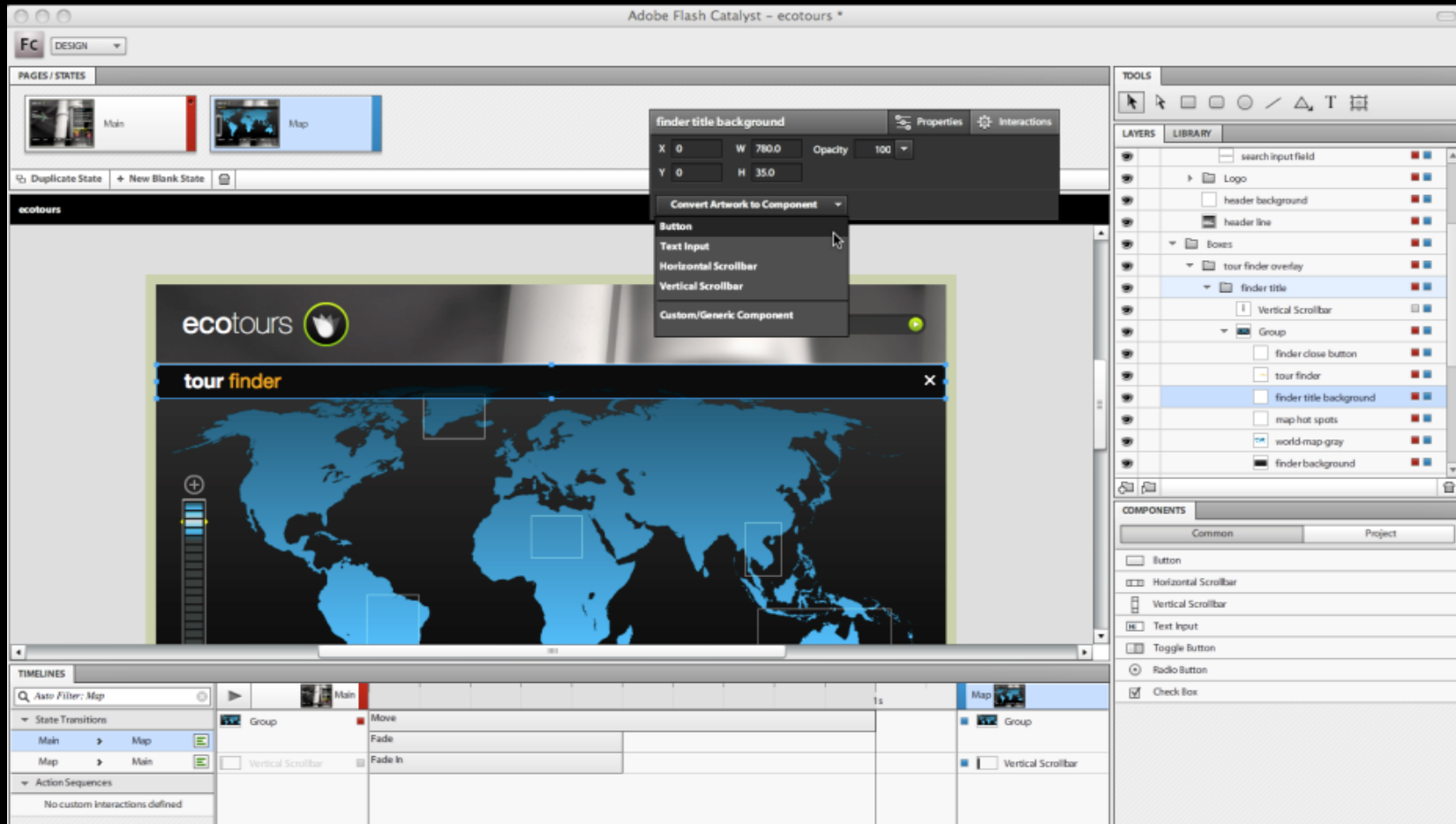


交互设计师

Adobe Flash Catalyst – 第二个公开测试版本

FC

面向设计师的专业交互设计工具，无需额外编码即可快速创建Flex应用的用户界面和交互内容



目标用户

- 交互Web设计师/交互设计师

负责在视觉设计或界面上创作交互体验的设计人员。使用 Photoshop, Illustrator和Fireworks设计原型，并且更加注重设计交互行为的群体。

- 信息架构师

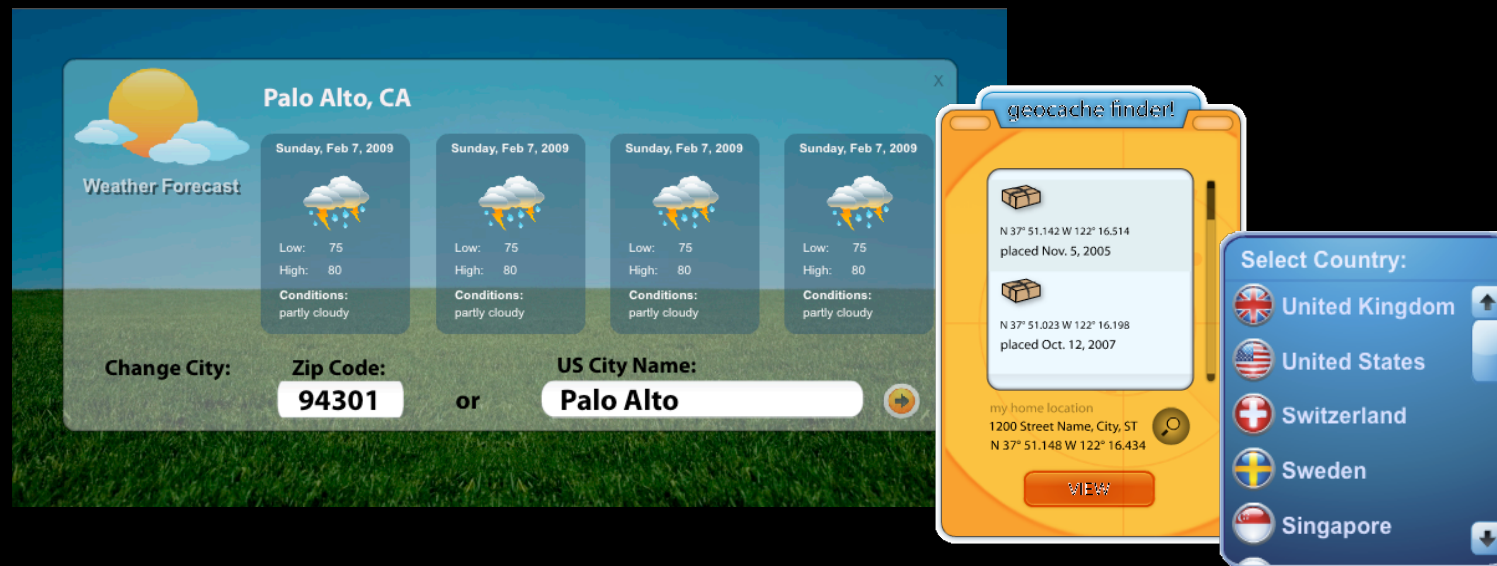
针对媒体元素和用户体验负责创建高级用户交互模型，负责早期的的交互流程和框架以及创意的实现过程制定。

- 图形设计师和创意总监

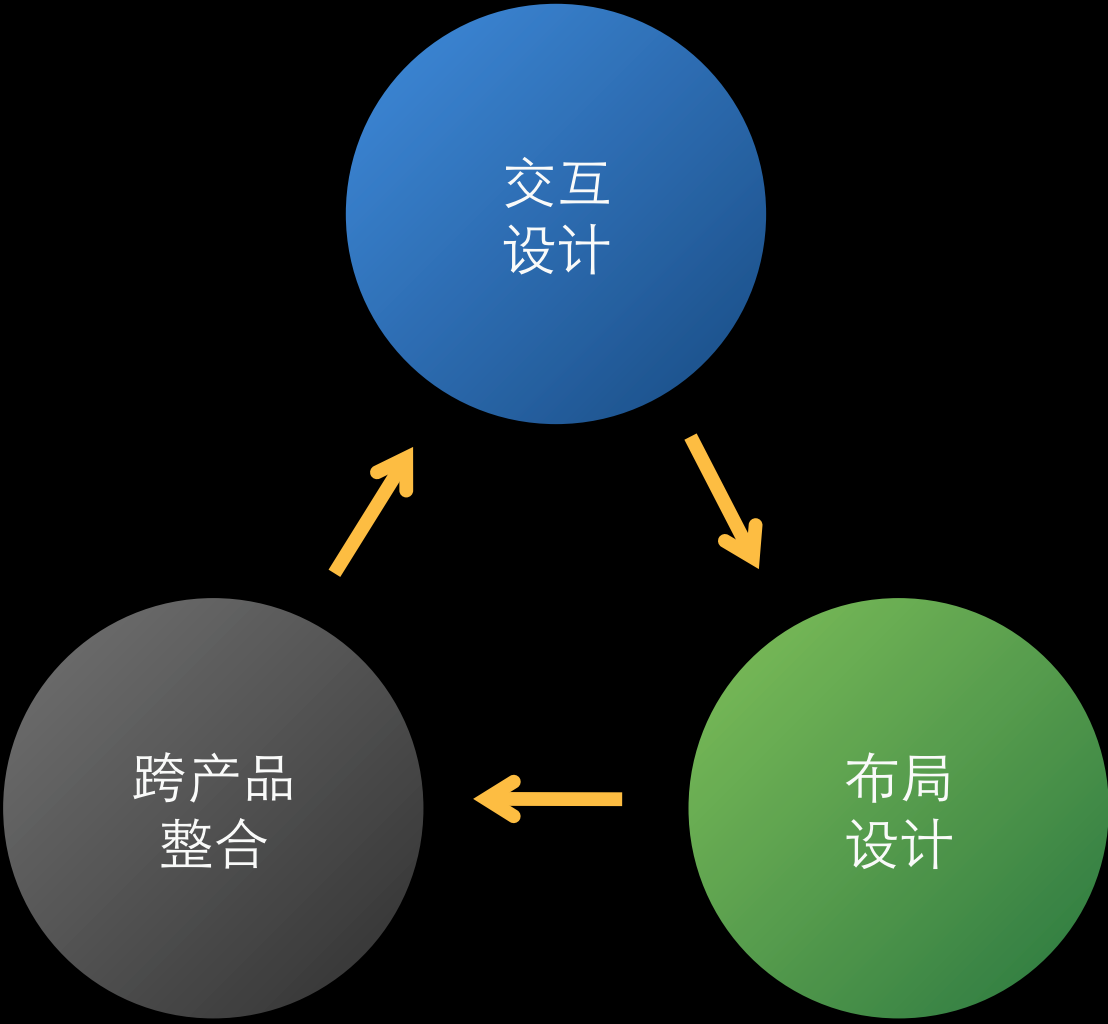
渴望不需要通过编码就实现交互过程或演示站点原型的人群

Flash Catalyst适用范围

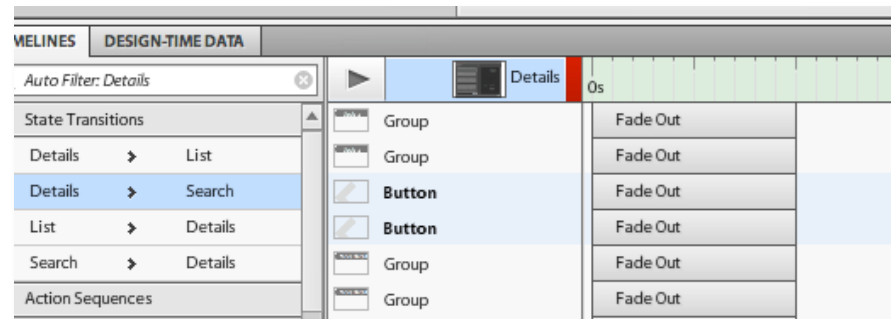
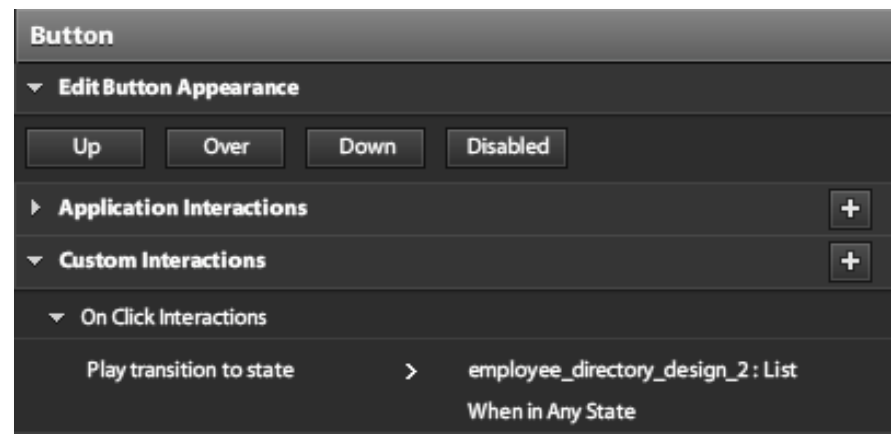
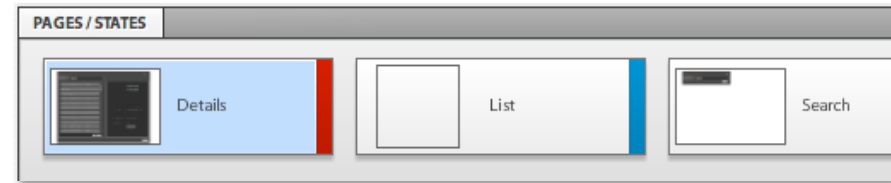
- 应用界面
- 交互内容
- 各种组合
- 产品演示手册
- 原型
- 基础UI框架
- 小型交互站点
- 网站导航
- 小应用
- 交互式教育内容



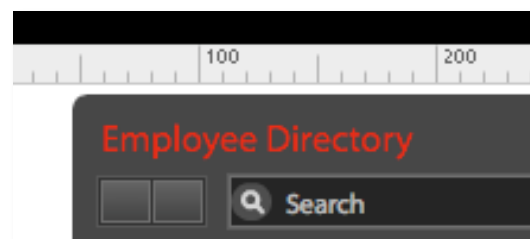
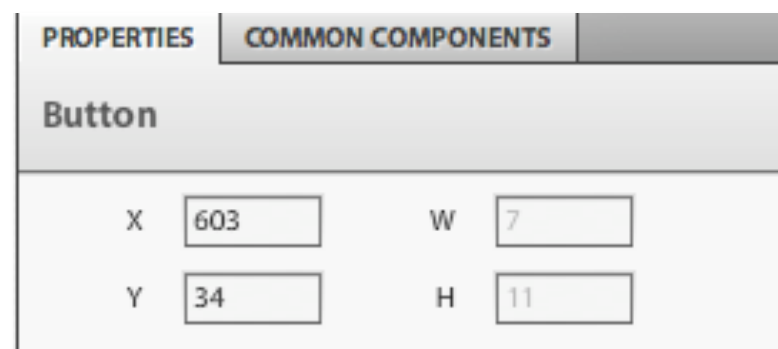
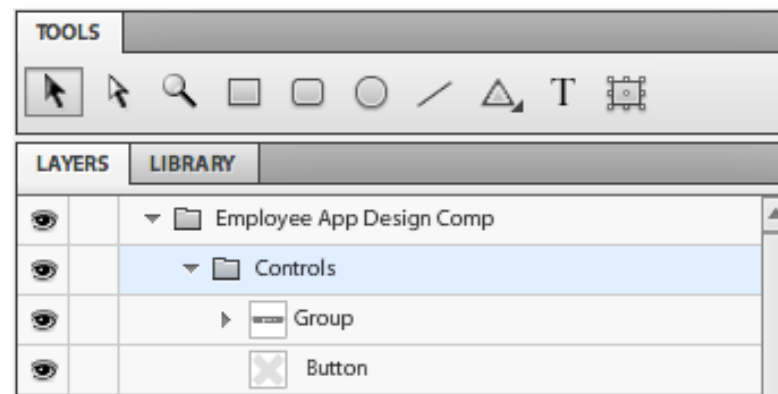
Flash Catalyst 功能



1. 通过Pages管理项目，通过States管理可视化对象
2. 不需要编码进行交互设计
3. 将静态设计元素转换为动态组件
4. 从设计元素创建数据list
5. 设计环境下的模拟数据显示
6. 组件扩展性
7. 交互式HUD面板 (Heads-up Display智能显示面板)
8. Library面板
9. States面板
10. 转换特效和行为顺序编辑
11. 时间轴面板



1. Artboard 绘画板（工作区）
2. 像素级精确控制元件位置
3. 层面板
4. 属性面板
5. 矢量图绘制工具
6. 布局设计高效辅助工具



1. 与Photoshop 和 Illustrator整合
2. 与Flash Builder整合
3. 动态媒体支持
4. Flash Player 10 支持
5. 在浏览器中预览项目
6. 输出到Flash Player
7. Flex 4 framework 支持
8. Windows and MacOS (Intel) 支持
9. 原生FXG格式



Demo

A large, dark, 3D-style 'FC' logo centered on a light gray square background. The letters are bold and have a slight shadow, giving them a three-dimensional appearance.



Flash Builder 4

Flex Builder 3
之上的新功能

A large, dark, 3D-style logo consisting of the letters 'F' and 'B' in a bold, sans-serif font. The letters are set against a light gray, gradient background that is part of a larger square graphic on the right side of the slide.

更多统一的产品名称

FLEX 3 一代

FLEX 4 一代



Flex framework,
Flex Builder



Flash
Builder



Flash
Catalyst

统一工具的品牌命名

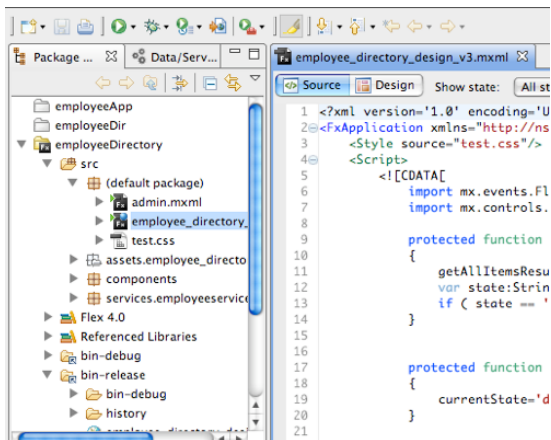


Flex framework

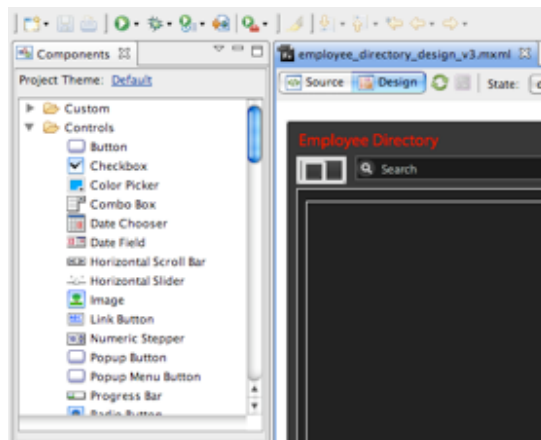
将Flex的品牌专注在开源框架之上，通过Flash Builder和Flash Catalyst去实现应用设计开发

什么是Flash Builder 4 ?

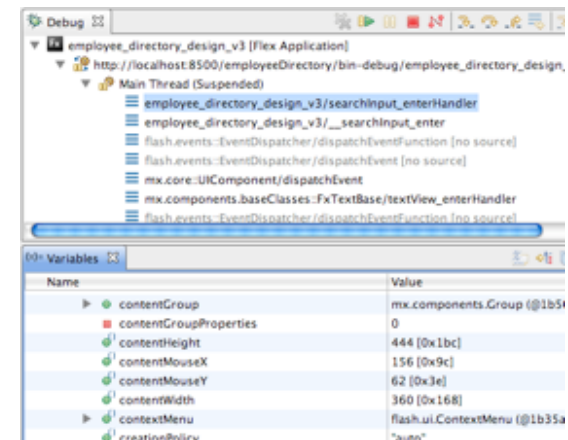
一个专业级的，基于**Eclipse IDE**环境设计的工具软件，用于帮助开发者快速构建跨平台的丰富互联网应用内容



MXML ActionScript
代码编辑



可视化布局 and 样式



交互式调试和分析

实现Flex应用的途径

- 应用有代表性地始于设计或数据逻辑

专注设计



- 信息架构考量
- CS工具套装使用
- 特别重视外观和创意设计

专注数据逻辑



- 数据驱动或服务模型
- 面向任务机制（添加或更新数据库记录）
- 程序可用性第一

实现Flex应用的途径

- 不论出发点如何，实现和测试代码阶段都类似

DESIGN FOCUSED



DATA FOCUSED



编码

- 手写
- 维护
- 代码格式化

测试和部署

- 分析
- 调试
- 构造

在一个生产环境下既支持设计又支持面向数据逻辑开发

设计师/开发者

- 设计稿保真
- 简化工作流程
- 避免人工操作步骤

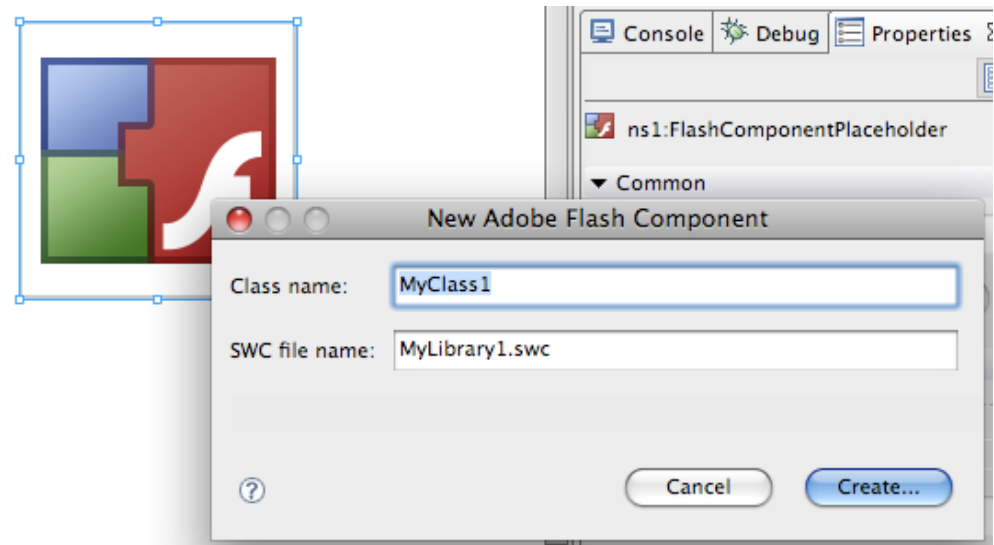
数据为中心的开发

- 支持多种不同的后台技术
- 简易化管理数据操作
- 生成数据绑定UI

IDE效率及测试

- 增强手写代码的辅助功能
- 加速调整和调试部分
- 适应已经存在的测试和构建过程

- Adobe Flash Catalyst 工作流
 - 在Builder中打开Catalyst FXP项目
 - 新的states状态编辑器
- Adobe Flash Professional 工作流
 - 全新的“Flash 组件”
 - 打开Flash Professional, 创建和编辑组件后, 返回Builder继续编辑
- 主题支持
 - 全新的UI应用主题方式
 - 更易于管理应用外观和组件样式



1. 定义服务

- ColdFusion
- PHP
- Java
- SOAP
- REST

2. 服务模型

- Flash Builder 自动检查服务
- 构造设计数据模型
- 配置数据类型和操作方法

3. 给Flex UI组件绑定数据

- 数据绑定
- UI 生成
- 分页
- 数据管理

- ColdFusion

- 调用任何ColdFusion Component (CFC)
- ColdFusion Builder有新的工作流程来生成数据访问的CFC组件



- PHP

- 内置 Zend Framework支持
- 使用Zend 组件或者任何PHP5类
- 与Zend Studio的新工作流, 用于生成类



- Java

- BlazeDS
- LiveCycle Data Services ES
- LiveCycle Model Editor



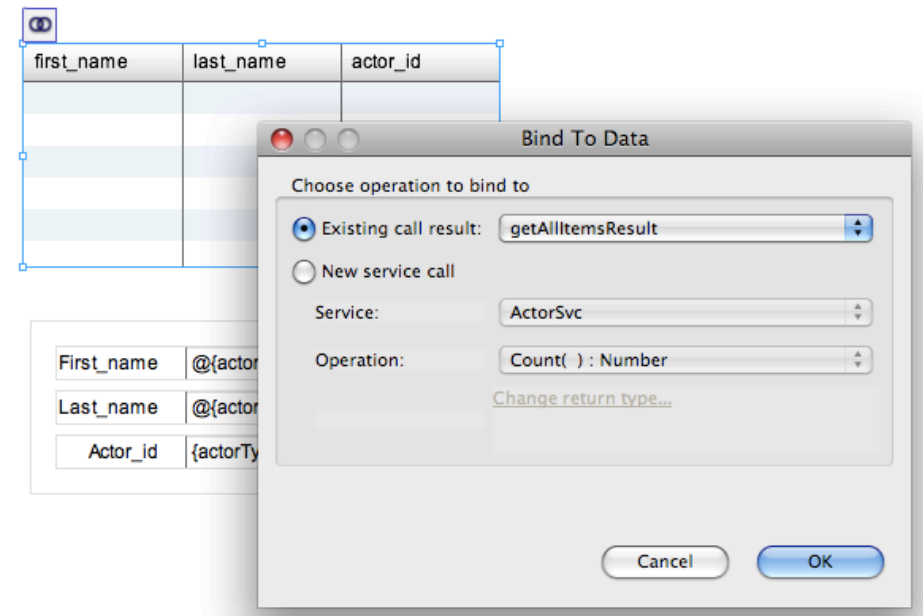
- HTTP, REST, SOAP Web Services

- 可扩展- 可以针对外部数据服务进行支持

<WSDL>

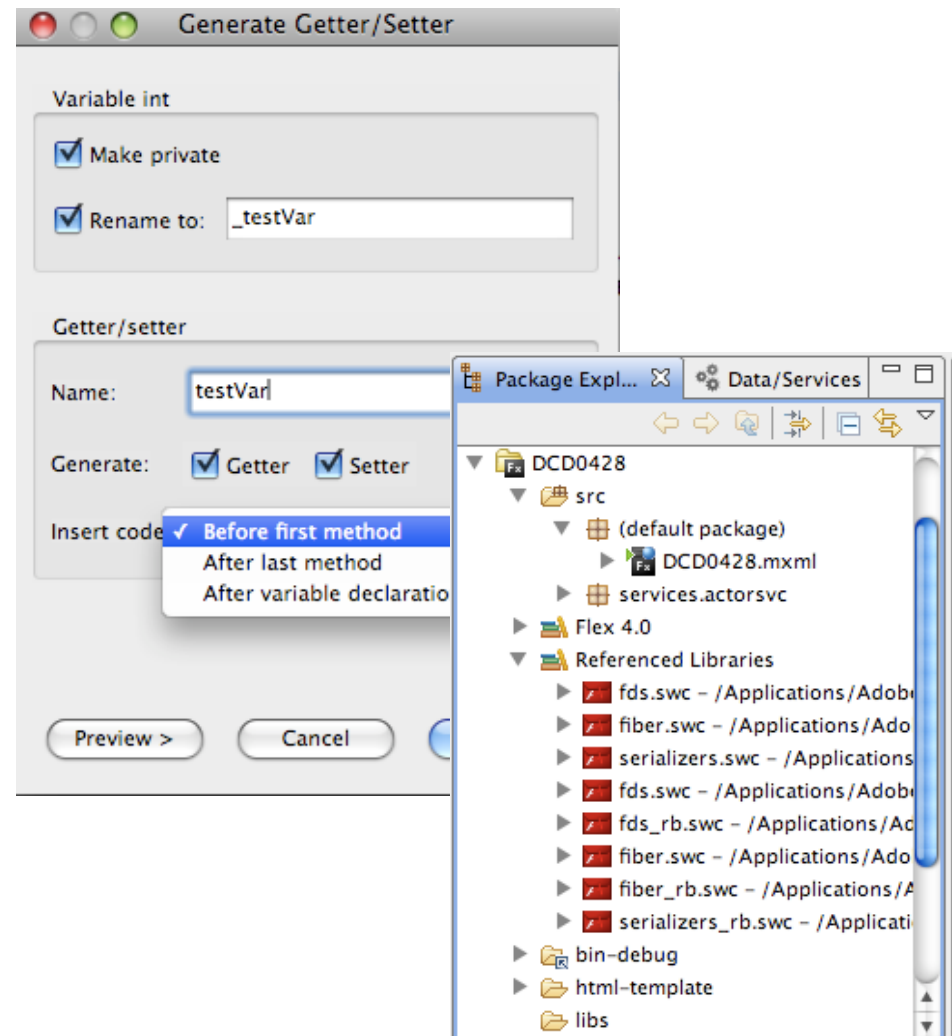
数据为中心的应用开发：数据绑定UI

- 将数据服务操作结果拖拽绑定到UI之上
- 从组件事件调用操作方法 (e.g. button clicks)
- 生成主要/详细记录关系
- 生成图表



- 映射客户端数据操作到服务器端的操作
 - 与Lifecycle DataService 3配合
 - 创建，更新，删除，等等
 - 自动大量更新和删除操作
 - 支持数据变化引发的“回滚”操作
- 任意记录的一个实例对象
 - 不论输入在何处显示，变更映射持续发生
- 大型数据记录集的分页
 - 自动化获取记录集中的若干条记录
 - 完美的针对大型数据集合在grid和list中的处理方式

- 包浏览器
- Getter/setter 生成向导
- 代码自动化缩进
- 自定义文件模板
- ASDoc 内置整合查询



- 网络监控
- 单元测试
- 代码定位和覆盖
- 命令行构建

Monitor is recording

Service	Operation	URL	Request time	Re...(B)	Resp...e time	Re...(B)	Elaps...e(ms)
RemoteService	getAllItems	http://localhost:8500/flex2gateway	08:54:29	304	08:54:29	10,540	100
RemoteService	getAllItems	http://localhost:8500/flex2gateway	08:54:06	304	08:54:07	10,529	611

OK

Request Parameters Tree view

```
Status = OK Request sent
service = RemoteService
operation = getAllItems
params = null
bytes sent
  ID = 50C1ABCA-8FB6-2738-37E6-204A021291F8
  body = 280
  header = 24
```

Response Result Tree view

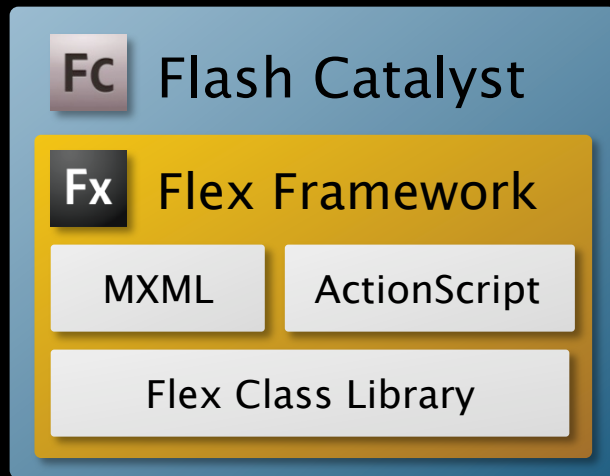
```
Status = OK
bytes received
  ID = 50C1ABCA-8FB6-2738-37E6-204A021291F8
  body = 10540
  header = 0
```



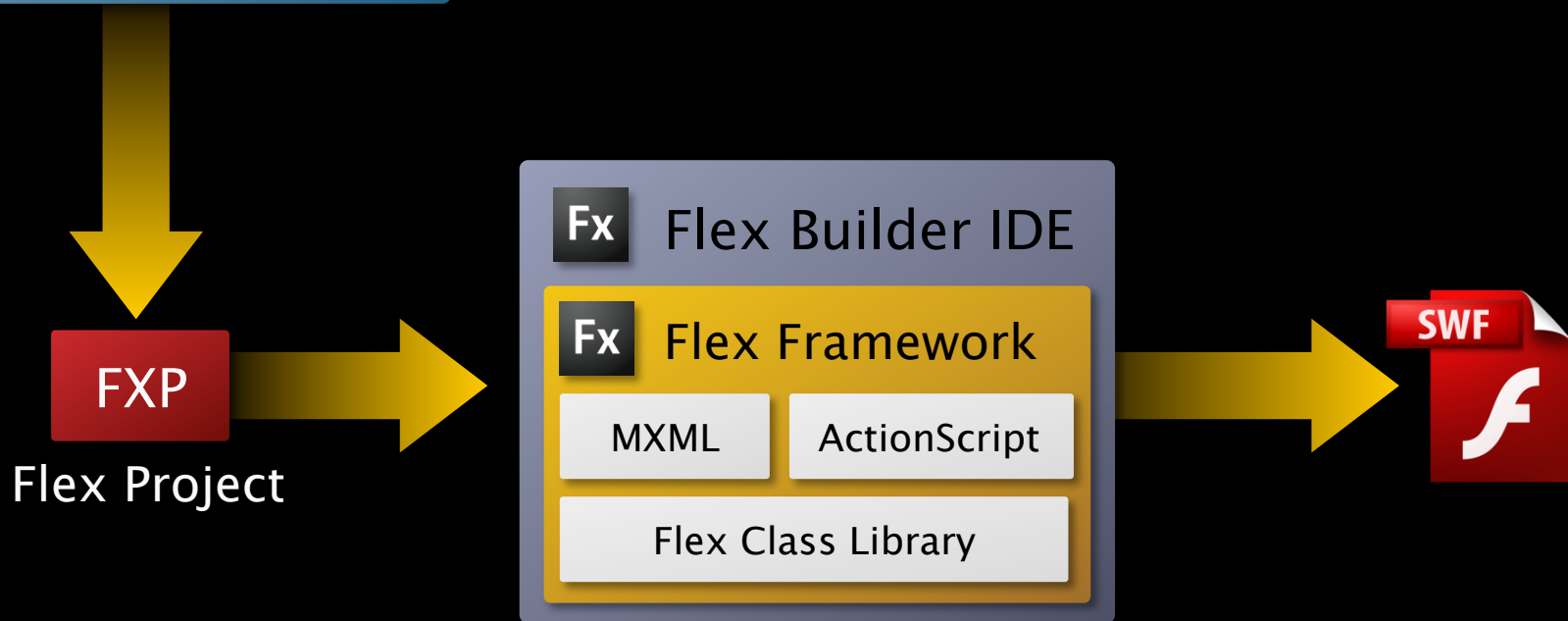
Demo

FB

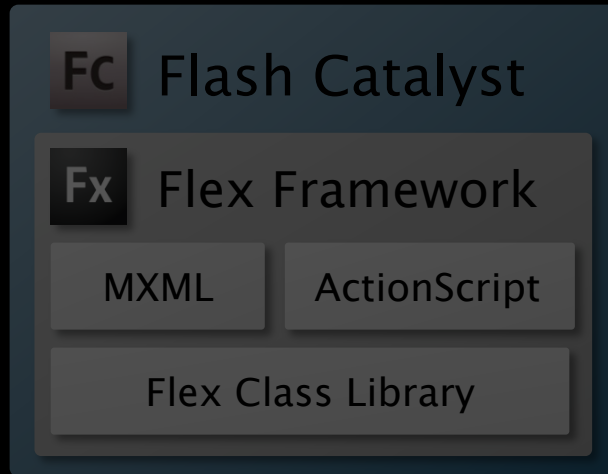
扩展Flex框架的角色



现在具备一个针对设计和开发通用统一的框架

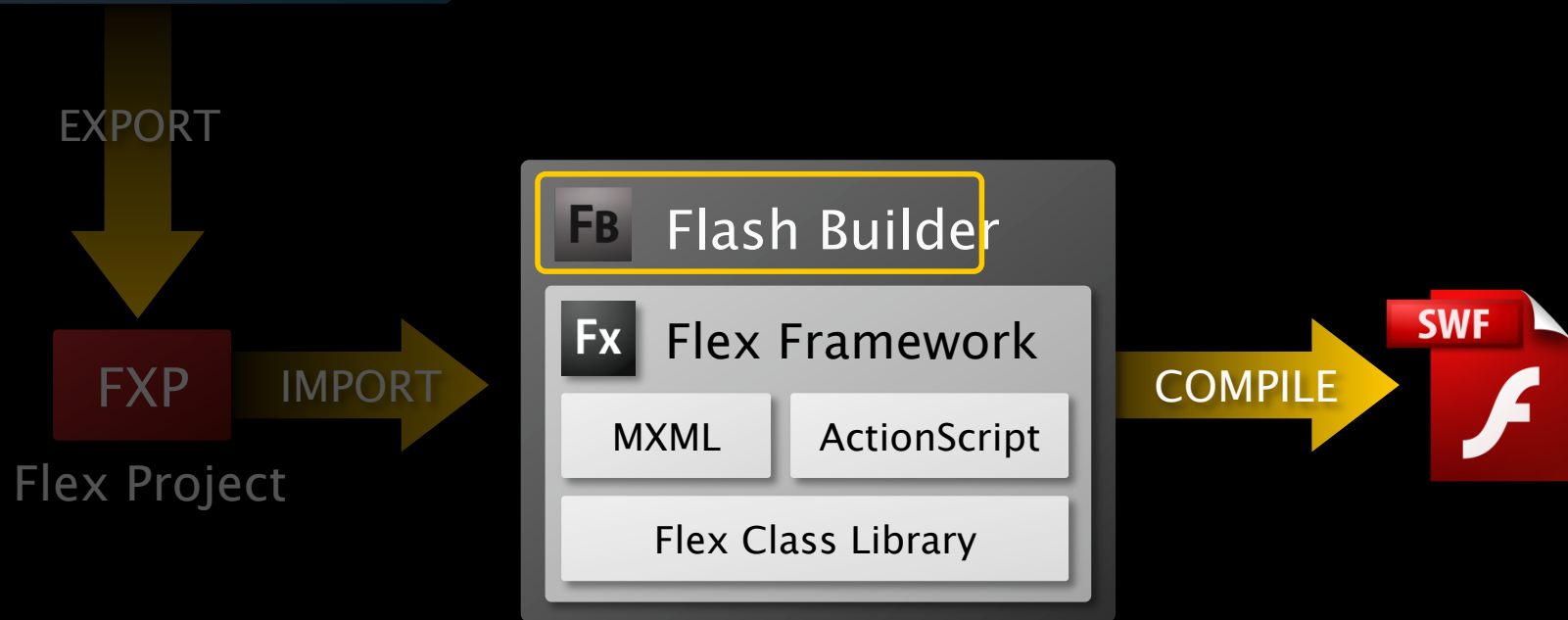


扩展Flash Builder的角色

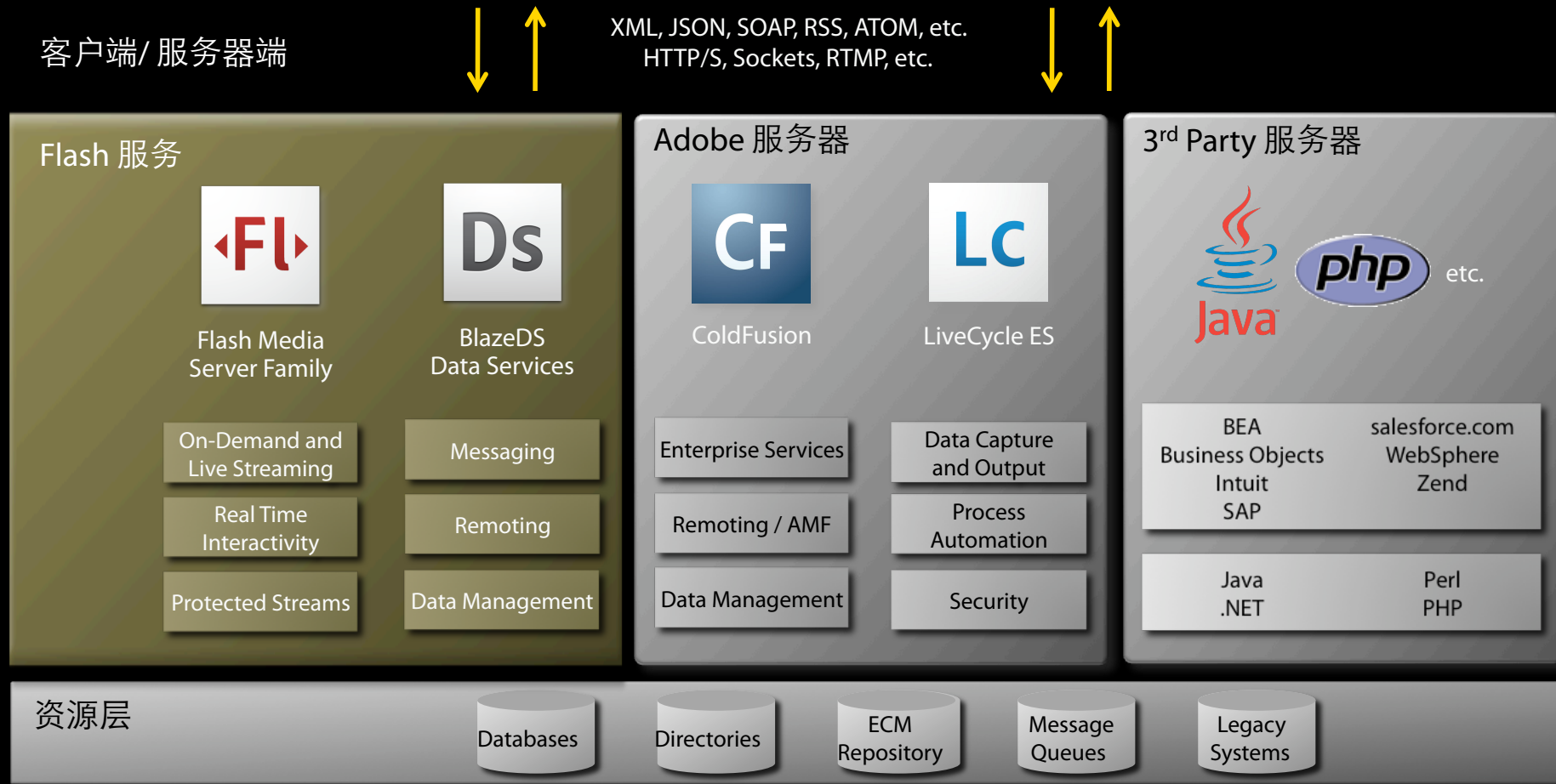


更改命名为Flash Builder:

- 针对纯粹的ActionScript 开发者
- 上升至使用Flash平台技术的相关开发
- 减少开源和商业软件之间的困惑



Adobe Flash Platform 架构: 服务器端



移动设备上的Flex框架 - Slider

The Flex logo, featuring the letters 'F' and 'x' in a bold, white, sans-serif font. The 'F' is significantly larger than the 'x'. The logo is centered within a dark gray square that has a subtle radial gradient.

Slider – Flex Mobile Framework

■ 目的

- 服务于未来云计算，社会化网络，连接互联网的各种设备的RIA应用
- 跨越多种设备显示屏幕，提供一个优化的通用RIA应用框架
- 将Flex扩展到未来的移动设备之上

■ 特点

- 使用现有Flex开发技巧，起源自Flex Framework
- 实现跨平台快速Flex开发
- 增强应用用户体验
- 简化开发定制流程，提高效率，降低成本

■ 需求

- 各种显示屏幕的移动设备，尤其是Smart Phone
- 400Mhz
- 128MB RAM

Slider – Flex Mobile Framework

■ 功能特点

- 更加利于使用工具完成针对不同屏幕尺寸的应用开发
- 布局适应性更加出色
- 提供原生APIs支持开发者操作顶级屏幕导航区
- 提供APIs允许开发者管理应用的历史和屏幕显示顺序关系
- 提供针对设备的自动化内存管理
 - 例如，应用处于限制状态时，内存的占用优化和资源的合理分配
- 更加适合移动设备应用使用的组件库，比如按钮，方向键位交互导航等
- 设计原则更适合UI设计
- 全面支持Flash CS5，可运行于iPhone应用之上

■ 时间表

- 2010年

Slider – Flex Mobile Framework

PHOTOSHOP.COM MOBILE APP FOR iPhone™





Q&A



ADOBE® FLASH® PLATFORM