

معرفی Actor Based Programming و توسعه نرم افزار های مقیاس پذیر و دارای عملیات همزمان بسیار زیاد -  
قسمت دوم

عنوان:

ایمان رحیمی نیا

نویسنده:

۱۴:۴۵ ۱۳۹۴/۰۵/۲۲

تاریخ:

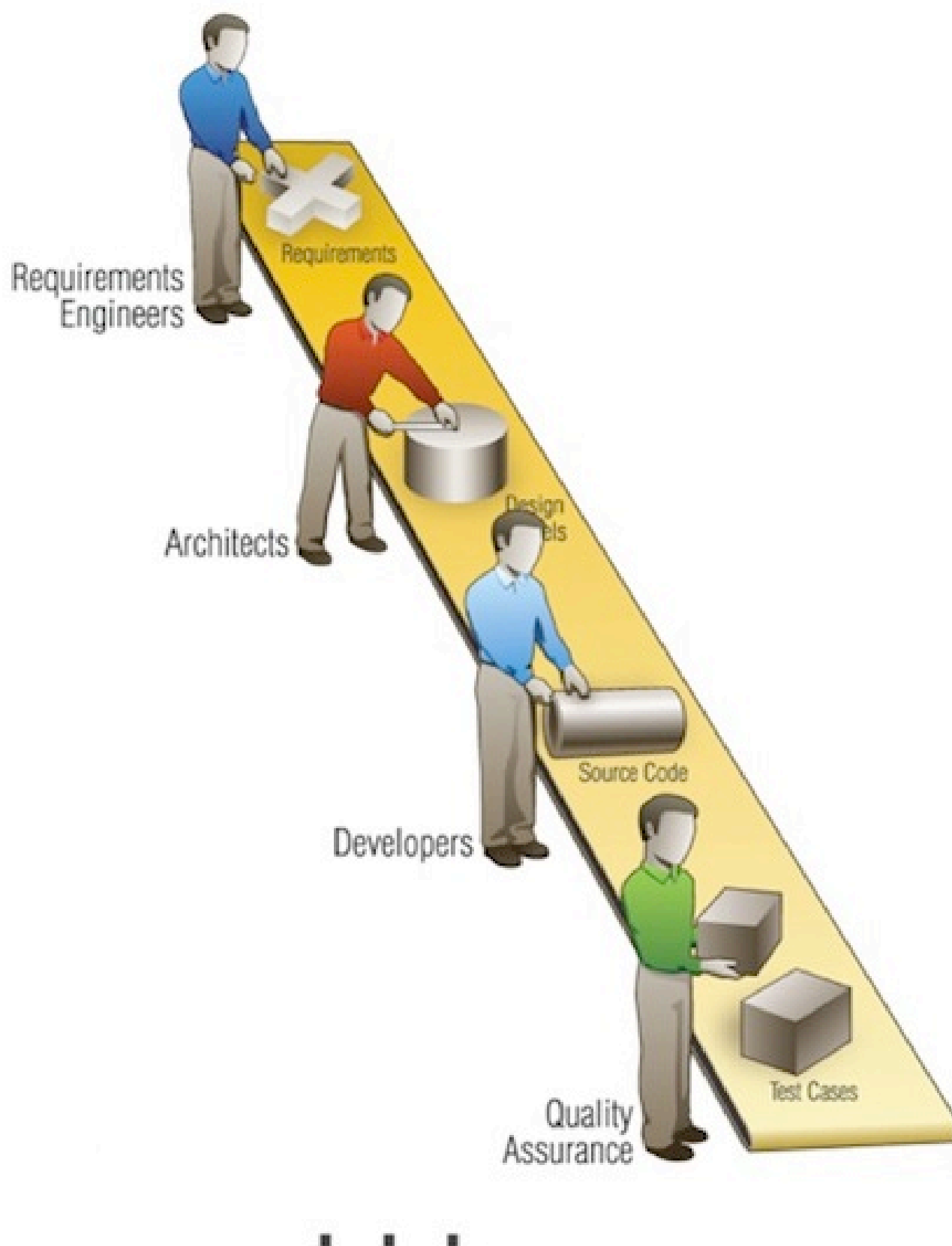
[www.dotnettips.info](http://www.dotnettips.info)

آدرس:

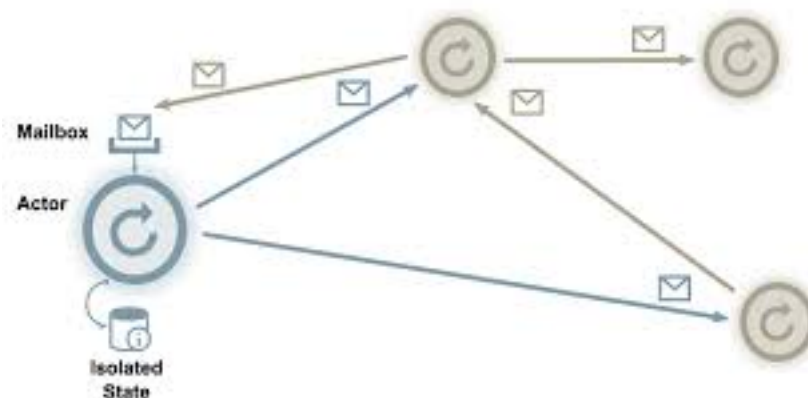
Asynchronous Programming, Tpl DataFlow, Erlang, Akka, actor based programming, TPL

گروه ها:

در [قسمت قبل](#) توضیحاتی راجع به مقدمات Actor Based Programming و کاربرد آن داده شد و چند framework نیز برای توسعه به این روش معرفی گردید. در این قسمت جزئیات بیشتری را از این روش توسعه، ارائه خواهیم داد. خط تولید کارخانه ای را فرض کنید که در آن یک قطعه از ابتدای خط حرکت نموده و کارگران مستقر در خط تولید نیز هر کدام بنا به وظیفه خود، کاری را بر روی قطعه ای مورد نظر انجام می دهند؛ به طوریکه در انتهای خط تولید، آن قطعه ای اولیه، به یک محصول کامل تبدیل می شود.



ایده‌ی Actor Based نیز هم از همین روش الهام گرفته است. با این تفاوت که بجای کارگران، Thread داریم و بجای قطعه نیز یک پیام یا object و بجای خط تولید نیز خط لوله یا pipeline را داریم. همانطور که در قسمت قبل اشاره کردم، وظیفه‌ی توسعه دهنده در این روش، طراحی یک خط لوله و نوشتن کد مربوط به هر thread است. به همین سادگی! یعنی تمام پیچیدگی‌های مربوط به concurrency و مسائل فنی توسط یک framework مثل TPL DataFlow یا Akka کنترل و مدیریت می‌شود و توسعه دهنده با تمرکز بر روی مسئله‌ی خود، شروع به طراحی (کانفیگ) خط لوله و نوشتن کد مربوط به هر کدام از threadها می‌نماید.



تصویر بالا یک خط لوله را با چهار اکتور، نشان می‌دهد. می‌توان اینطور فرض نمود که هر اکتور یک mailbox دارد و اگر پیامی برای آن اکتور بفرستید، آن را پردازش نموده و کار مخصوص به خود را بر روی آن پیام انجام می‌دهد و سپس آن پیام را برای اکتور بعدی خود ارسال می‌کند. اکتور دوم نیز به همان ترتیب کار خود را انجام داده و پیام را به اکتور مابعد خود ارسال می‌کند و به این ترتیب، یک پیام در خط لوله حرکت نموده و فرآیند مربوطه انجام می‌شود. اگر دقت کنید یک فرق دیگر هم بین خط تولید کارخانه و این خط لوله وجود دارد و آن این است که این خط لوله به صورت گراف می‌باشد. یعنی اکتورها می‌توانند در ارتباط خود یک حلقه را تشکیل دهند و یا یک اکتور با چندین اکتور ارتباط مستقیم داشته باشد (مثل اکتور سمت چپ تصویر که با دو اکتور دیگر در ارتباط است).

خوب حالا که با مفاهیم خط لوله و اکتور آشنا شدیم، یک مسئله‌ی بسیار ساده را در نظر می‌گیریم و آن را با این روش حل می‌کنیم. فرض کنید یک رشته (string) داریم و می‌خواهیم عملیات زیر را بر روی آن به ترتیب انجام دهیم:

1- فاصله‌های اضافی ابتدا و انتهای رشته حذف شود.

2- اگر رشته یک کلمه‌ای است lowerCase شود.

3- اگر رشته بیش از یک کلمه است، تمام کلمات، به جز کلمه‌ی اول، حذف شوند و سپس مرحله‌ی 2 بر روی آن انجام شود.

4- نتیجه‌ی کار در خروجی نمایش داده شود.

حالا می‌خواهیم انجام هر یک از عملیات فوق را به یک اکتور سپرده و یک خط لوله را برای حل این مسئله طراحی کنیم. در قسمت بعدی به صورت عملی و با TPL DataFlow مایکروسافت این کار را انجام می‌دهیم.