

조직을 변화시키는 패턴 이야기

EVA네가 들려주는 Fearless Change 첫 번째 이야기

자신이 몸담고 있는 조직이 능동적이며, 합리적이라고 생각하는가? 만약 “예”라고 생각하면 정말 좋은 회사에 다니고 있는 것이다. 하지만 여러분의 제안이 별 검토 없이 거절되거나, 제한한 사람에게 일을 맡겨 버려, 더 이상 어떠한 의견 공유도 이루어지지 않는 조직이라면 어떻게 해야 할까? 경력이 쌓이다 보면, 기술보다 사람들과의 협상과 대화를 통해 기회를 만들 수 있느냐가 더 중요하게 와 닿을 것이다. 이번 글에서는 조직을 점진적으로 변화시키고, 능동, 자발적으로 만드는 패턴들을 소개하고자 한다.



EVA네 (정리 · 손영수) arload@live.com (아키텍트로 가는 길) | 한국에 몇 안 되는 패턴 저자이며, 데브피아 소프트웨어공학 스터디인 EvaCast를 7년째 이끌고 있다. 이 땅의 개발자들을 위해 스터디 멤버들과 함께 한 결과물을 <http://www.EvaCast.net>을 통해 무료로 공유하고 있다. 부족한 실력이지만 지식을 나눌 때는 누구보다 '부자'라는 자부심을 가지고 지식 나눔에 힘쓰고 있다. Pattern 전도사를 꿈꾸고 있으며 PLoP과 같은 Pattern 학회를 국내에 만드는 것이 꿈이다.

PLoP(패턴 학회)에 참여하면서 그 어느 컨설턴트보다 깊은 감명과 새로운 시선을 준 건, 바로 Linda Rising이었다. 백발의 열정적인 그녀의 모습과 구성원들 간의 마찰을 부드럽게 다듬고, 조직을 서서히 변화시키는 그녀의 행동은 아키텍트를 꿈꾸는 필자에게 많은 것을 깨닫게 하기에 충분했다. 이번 글은 소프트웨어 설계 스터디 그룹인 EVA 네에서 나온 얘기들을 모아 독자들과 공유하는 것이다.

아이디어가 좋다고 무조건 성공하는가?

좋은 아이디어가 사장되는 원인으로 우리가 생각하는 몇 가지 오해들이 있다.

첫째, 좋은 IDEA이기 때문에 혁신이 그냥 받아 들여 질 거라고 생각한다. 그렇다면 왜 Sony의 Beta 방식이 VHS에 졌을까?, Mac OS가 왜 MS DOS에 졌을까?, IDEA도 중요하지만 개방성과 범용성과 같은 다른 부분들도 고려해야 된다는 게 단적으로 드러난 예다.

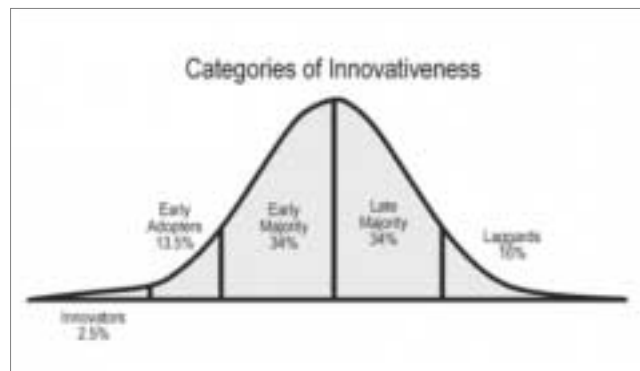
둘째, 새로운 좋은 아이디어가 소개되면, 별 노력 없이 사용될 것이라는 생각 때문이다. 새싹을 심어놓고 물을 주지 않으면, 과연 잘 자라날까? 끊임없이 인내심을 가지고 전파를 해야 된다는 것이다. 만약 직급을 이용해 Top-Down형태대로 위에서 눌러 찍으면 빨리 아이디어가 전파되겠지만, 여러 가지 저항이 발생할 수도 있다. 이와 반대로 Bottom-Up 형태로 점진적으로, 목적의식을 심어주고 방향성을 제공함으로써 구성원들의 자발적인 변화를 이끌어 내는 방법도 있지만 실득과 소통으로 인해 변화가

상대적으로 느리게 전파된다.

그럼 부드럽게 변화를 주도하고 이끄는 방법은 무엇이 있을까? Linda Rising의 Fearless Change 패턴은 Top-Down, Bottom-Up적인 변화를 조화롭게 사용하여 조직에 변화를 이끌어 내는 패턴이다.

변화를 거부하는 조직 문화와 그 속에 피어나는 민들레

우리는 먼저 조직 문화가 변화를 쉽게 수용할 수 있게 만들어야 한다. 한 목수가 나무 하나를 자르는데 5시간이나 걸렸는데, 그 이유는 너무 바빠 도끼 날을 갈 시간이 없어서였다고 한다. 너무 바빠 품질과 효율성 향상시키기 위한 방법이나 교육들을 배울 시간이 없다는 핑계로 오히려 더 안 좋은 결과를 만든 상황이다. 한국의 많은 회사의 문화가 이렇지 않을까? 교육 하나 신청하기도 힘들고, 심지어 교육을 가면 나오는 효율성을 증명해야



〈그림 1〉 Roger의 Innovation Curve

보내주겠다는 회사도 있다. 이런 조직에서 변화를 이끌어낸다는 것은 매우 어려운 일이다. 먼저 변화를 수용하는 사람의 유형을 파악해 보자. 사람들이 변화를 받아들이는 성향을 그룹화한 Roger's Innovation Curve라는 것이 있다. 여기서는 변화를 받아들이는 그룹을 5가지로 구분한다.

- Innovators - 변화를 이끌어가는 혁신적인 사람들
- Early Adopters - 새로운 아이디어 채택을 다른이 보다 일찍 조심스럽게 수용하는 사람들
- Early Majority - 신중한 사람들, 조심스러우나 일반적인 사람들보다 좀 더 빨리 변화를 수용
- Late Majority - 회의적인 사람들, 대다수의 사람들이 새로운 아이디어와 제품을 사용할 때 사용
- Laggards - 전통적인 사람들, 옛방식을 고수하는 것을 좋아하고, 새로운 아이디어에 비판적이며, 새로운 아이디어가 주류가 되거나 심지어 전통이 되어야 수용

이 Curve에서 알려 주듯이, 새로운 것을 좋아하는 진보적인 성향을 가진 사람부터, 점진적으로 설득하면서 변화를 이끄는 것이 좋은 방법일 것이다.

어디서부터 변화를 시작해야 하나?

그건 바로 마음가짐이다. 내 자신이 변화를 전파하는 Evangelist가 되어야 하며, 열정을 지속적으로 유지할 수 있는 Dedicated Champion(헌신적인 챔피언)이 되는 것이다. 성공적인 Dedicated Champion이 되기 위해선 변화를 주도하는 역할 역시 업무의 부분으로 인정받아야 된다. 하지만 대부분의 회사가 어떠한 가시적 성과가 나오는데 업무의 초점이 맞추어져 있다는 것이 문제이다. 그리고 긍정적으로 변화하고 있다는 것을 보이기 위해서 무언가 지속적으로 일이 잘 진행된다는 것을 외부에 알려야 하며 또한 잘 평가 받기 위해서 근거와 자료를 꾸준히 만드는 작업이 필요하다. 높은 상사에게는 프로세스에 대한 도입으로 반복되는 실수나, 문제점들이 서서히 줄었다고 지표로 뽑아내어 예전보다 나아졌다고 평가(하지만 변화를 지지하는 높은 상관에게 지속적인 지원을 받기 위해서는 이러한 평가자료도 매우 중요하다.)를 하는 것만큼, 모두다 변화의 장점을 인식하고, 이것이 긍정적으로 바뀌어 현재 잘 적용되어 있다고 팀원들이 느끼는 것도 중요하다.

먼저 식사나 간식 때 살짝 떠 보세요

먼저 Do Food 패턴과 Brown Bag 패턴을 이용해 회의라는

무거운 형태의 모임이 아닌 가벼운 모임의 형태로 대화를 시작해서 나의 의견에 동조하는 사람을 찾아야 된다. Do Food는 음식을 나누면서 얘기하는 패턴으로, 음식이 상당히 분위기를 부드럽게 해 주고 아이디어를 샘 솟게 해 준다는 것이다. 여기서 주의할 것은 특정인이 먹지 못하는 음식이 있을 수도 있으니 몇몇 종류의 음식을 준비하는 것이다. 그리고 Brown Bag은 우리가 샌드위치 집에서 보는 갈색의 종이 백을 기억하면 될 것이다. 바로 이러한 가벼운 음식을 먹으면서 식사 시간에 대화를 나누는 것이다. 회의의 무거운 형태가 아닌 식사 시간의 가벼운 대화만으로도 나의 의견(변화 내용)에 동조하는 사람을 찾을 수 있다.



〈그림 2〉
뜨거운 물인지
차가운 물인지 검사해라

이때 사용되는 패턴이 Test the water다. Test the water의 의미는 나의 의견에 동조하는지 보고, 물에 발을 담근다는 정도로 생각하면 된다(나의 얘기에 동조하는 사람을 찾아내는 패턴). 가벼운 주제로 이러한 변화에 대해서 어떻게 생각하느냐 얘기를 나누는 것만으로 충분하다. 또한 변화가 어려운 조직은 너무 처음부터 큰 변화를 주도하기 보다는 조금씩 변화를 주는 것이 중요하다. Small Success의 또 다른 얘기겠지만 S군이나, H님(EVA 스티디 멤버)을 비롯한 몇몇 분의 조언처럼 사내 메일을 통해서 새롭고 유익한 정보를 계속해서 공유해 나간다. 그래서 자신의 얘거나 메시지를 숨겨서 계속 전달하면 자연스럽게 생각을 전파시킬 수 있다. 잠시간 메일링 리스트를 중지함으로써 그 효과를 바로 알 수 있다. 누가 이런 유익한 정보를 전달하는 것을 그만두었냐며, 바로 피드백이 온 사례들을 많이 보았다고 한다. 그리고 아예 이러한 것을 업무의 일부분으로 할당 받는 사람들도 있었다.

또한 H님 같은 경우는 지금은 보험을 탈퇴했지만, 그 보험 설계사는 아직도 메일을 통해 끊임없이 유용한 소식이나 정보를 보내 호감을 갖게 되었다고 한다. 아마 다음에 보험을 들 기회가 있으면 꼭 이 보험 설계사에게 들어야겠다는 생각이 들 정도였다고 한다. 그리고 이러한 정보 메일이 원인이 된 것인지 모르겠지만 이 보험사의 연봉은 1억을 상회한다고 한다.

여기서 우리가 중요시 생각해 볼 것은 변화를 어떻게 전달했느냐는 것이다. 변화를 받아들이는 사람이 상식의 선을 넘지 않는 변

화부터 시작해야 된다는 것이다. 처음부터 급격한 변화는 누구나 받아들이기 힘들어 저항이 발생할 수 있다는 점을 기억해야 한다. 또 다른 변화의 예시로 필자의 경우 어떠한 변화를 다수결이 동의했음에도 불구하고, 소수의 높은 직급을 가진 사람의 반대로 좌절된 경우를 간간히 목격했다. 그 변화에 대한 안 좋은 결과가 발생하면, 그 책임을 지겠냐며 으름장을 놓거나, 좋음에도 귀찮은 작업이므로 변화를 거부하는 경우도 있었다. 직급 체계라는 구조 때문에 아래에서 위로 가는(Bottom-Up) 변화가 한국 문화에서는 전파되기 힘들다. 그러니 변화를 주도하기 이전에 조직 문화 등을 알 필요가 있다. 변화를 전파하기 이전에 과연 이 변화를 수용할 수 있을 건지 미리 설명하고 검토해야 된다는 것이고, 또한 변화로 인해 불이익을 받는 사람들을 파악해서 도움이나 다른 보상을 주는 것도 매우 중요하다.

변화는 시간이 필요하다

그 다음 우리가 가져야 하는 마음 가짐은 변화하기 위해 시간이 필요(Time for reflection)하다는 것이다. ‘온고지신(溫故知新)’ 처럼 옛 것을 기반으로 더 나은 것을 차츰 개선해 나가는 것이 중요하다. 회고를 통해서 더 나은 것을 만들 수 있다는 의미다. 기존 시스템의 문제를 파악해서 원인을 찾아내고 개선해 나가라. 다만 급격한 변화가 아닌 조금씩 바꾸어 나가는 것이 중요하다. 처음부터 큰 변화는 많은 저항과 함께 지속적인 변화에 적합하지 않다. 그렇기 때문에 작은 변화에 대해 서로 평가하고, 이 성공(small success)을 서로 공유해야 한다. 바로 작은 성공을 팀원들과 공유함으로써 지속성을 유지하는 것이 중요하다. 즉 작은 성공이 모여서 큰 성공을 이룬다는 절대 조건에 기반을 두고 있다. XP, Scrum에서 말하듯이 짧은 주기의 데모시간을 주어서, 그 사이에 성공을 공유하고, 잘못된 것을 바로 잡는 시간이 필요하다. 이렇게 하면 팀원들은 Lesson Learn하면서 올바른 형태로 프로젝트의 방향을 잡아갈 수 있다.

또한 프로젝트 도중 회고(retrospective), 검시(postmortem), 산후(postpartum) 또는 프로젝트 리뷰(Project review)를 통해 계속해서 Lesson Learn하는 자세 역시 중요하다. 또한 조직 간의 정치나 저항이 덜 발생할 수 있는 구조로 팀을 구성해야 한다. 대부분의 변화는 시나리오로 또는 Feature의 변화를 가져오는데 이러한 변화의 빈도에 따라 팀 역시 시나리오나 Feature 별로 구성되어야 한다.

이 이야기는 마소 2008년 9월호에 기고한 “팀 생산성을 높이는 이야기”의 땅콩버터와 마친루(<http://arload.wordpress.com/2008/05/20/peanutbutter/>)를 읽으면 이해가 될 것이다. Feature Creep이라는 스물스물 추가되는 변화를 받아들이기 위

해서는 변화를 잘 받아들일 수 있는 구조로 팀을 만들어 가는 것이 중요하다. Microsoft의 Feature Crew가 바로 좋은 예이다. 우리가 이 팀 구조의 대부분은 Architect 팀, 개발 팀, QA 팀을 따로 두었기 때문에 이 팀 간의 정치나 서로 간의 불신이 매우 크며, 그로 인해 변화를 받아 들이는 속도도 매우 낮아졌다. 그래서 Microsoft는 Feature 별로 팀을 나누는 새로운 시도를 했다. Architect와 PM은 전체 Feature를 서로 조율하는 역할을 하고, 개발자 3 : QA 7의 비율로 하나의 Feature를 아주 빨리 개발한다. 그렇기 때문에 이들 팀원 간에 의사 소통도 매우 잘 되었으며, 한 Feature를 불과 3~4주 만에 끝내고 빨리 다음 Feature를 개발할 수 있었다고 한다(만약 시나리오로 구성된 팀이라면 이것이 시나리오로 바뀌어도 별 차이가 없다).

대부분의 컨설턴트, PM이 실패하는 경우는 조직의 문화를 이해하지 않고, 자신의 철학이나 아이디어를 전파할 때 발생한다. 저항이 생기는 것은 당연한 일이다.

이러한 저항을 줄이기 위해선 PM이나 컨설턴트는 주기적으로 30분마다 하루에 몇 명과 커피나 차를 마시면서, 얘기를 나누고, 걱정거리, 근심거리를 해결해 주거나, 개발자의 취향이나 역량을 파악해서 그들이 원하는 일을 할 수 있는 배려를 하면 된다는 것을 EVA의 스터디 팀원인 현종님이 목격했다고 한다.

다른 이에게 도움을 요청해라(Ask for Help)

황우와 유방의 얘기에서 알 수 있듯이, 또한 미실의 말처럼, 사람을 적절하게 활용하는 것이 중요하다. 사람들에게 도움을 요청해라. 거절을 당할 수도 있지만, 거절을 당하더라도 다양한 사람의 여러 가지 조언을 들을 수도 있다. 도움을 요청하기 위해서는 끊임없이 Stay in touch(지속적으로 만나고 얘기를 나누는 패턴)해야 하며, 업무가 아닌 개인적인 얘기들도 공유할 수 있도록 친분을 쌓는 것도 중요하다. 1분씩 돌아가면서 서로에게 일어났



〈그림 3〉
도움을 요청하라

던 이야기나, 얘기들을 나누는 것도 좋다.

문제는 이러한 미팅에 절대 강제성을 부여해서는 안 된다는 점이다. 누군가가 돌아가면서 발표를 해야 한다면, 아침에 하는 것은 반대다. 이것 때문에 준비를 해야 된다는 부담감으로 다들 힘들게 준비하는 경우도 있고, 또한 아침 출석 체크와 동시에 상사의 명령 하달의 시간으로 변해서 오히려 자유로운 팀원의 분위기를 망칠 수도 있다. Brown Bag 패턴과 Do Food 패턴을 이용하라. 무조건적인 도움 요청으로 보일 정도로 도움을 요청해서도 안 된다. 잘 정돈된 형태로 질문을 해야 오는 답변 또한 명확해진다. 무엇에 도움을 받아야 할 지 정작 모르면서 현상만 얘기하면서 도움을 요청하는 경우는 오히려 반감을 불러 일으킬 수 있다. 충분한 검토와 생각 이후에 도움을 요청하는 것이 중요하다.

이젠 뭘 해야지? 사람을 얻어라

변화를 시작할 수 있다는 열정, 마음가짐을 가졌다면, 이제 조직에 변화를 가져올 수 있는 작업을 해야 한다. 바로 사람을 얻는 것이다. 나의 사람을 만들고, 같이 생각을 나누어 점진적으로 혁신의 생각을 전파해 나가는 것이다. 큰 전략은 다음과 같다. 인맥이 넓은 Connector를 통해 Guru를 만나 나의 아이디어와 생각을 다듬어 신뢰성을 확보하고 나의 상사인 Local Sponsor, 높게는 고위층인 Corporate Angel의 지지를 얻는다. 그리고 Connector를 통해서 Innovator 성격의 사람을 찾아내 변화를 만들어 내는 것이다.

Connector를 찾아라

내가 가진 아이디어를 조직 내에 모든 사람에게 전파하기는 힘들기 때문에 대인관계가 넓은 사람을 찾아가, 자신의 생각을 전달한 다음 그 사람을 통해 생각이나 아이디어가 다른 이에게 퍼져 나가게 해야 한다. 이때 주의해야 할 것은 집단의 성격과 Connector의 성격이 매우 유사하다는 것이다. 개혁적이고 진보적인 집단에서는 Connector 역시 개혁적인 성향을 가지고 있지만 보수적인 집단에서는 Connector 역시 보수적인 성향을 가진 경우가 많다. 즉 Connector는 그 집단을 대표하는 성격을 가질 가능성이 높다는 것이다.

전파를 할 사람의 성격, 성향을 먼저 파악한 후에 생각을 전달하라. 잘못하다가는 나의 생각이 부정적인 것으로 전파될 수도 있다.

Connector 중 외부의 변화를 잘 받아 들이는 사람은 토론과 청을 통해 설득을 해야 되고, 회의론적인 말을 잘하는 Connector는 호형호제, 또는 Do Food 패턴을 사용해서 친밀감을 느끼게 하고 설득하는 것이 좋다.

설득을 부드럽게 하는 방법

논쟁을 통해 상대방을 제압할 수는 있지만, 실제 그 사람을 설득할 만한 공감대를 같이 얻어내는 것은 힘들다. 논리로써 설득을 한 것처럼 보이지만 실제 더 큰 저항을 가져오는 경우가 많으므로, 대화(토론) 시 상대방과의 접점을 찾아내는 것이 중요하다. 그리고 EBS에서 실제 방영한 '설득의 비밀'이라는 방송에서는 16명의 사람이 가출한 아들 외에 여러 명을 설득하는 방법을 보여주었는데 6주의 트레이닝으로 다른 이를 세련되게 설득하는 방법을 전달해 주었다.

<http://tvpot.daum.net/>에서 '설득의 비밀'이라 검색하면 시간대가 긴 5개 동영상을 찾으면 볼 수 있으니, 시청하거나, 이 내용을 책으로 정리한 '설득의 비밀'을 읽어보길 권한다.

Just Say Thanks

조그만 일이 완료될 때에도 항상 격려나 고마움을 표현해야 된다. 카드나 아니면 조그만 선물이라도 나누면서 개인적으로 만나 감사함을 표현해 주는 것이 중요하다. 조그만 성공(small success)에 감사를 표현해 변화가 잘 발생되었다는 긍정적인 이미지를 모두와 공유하고 계속해서 변화를 주도하는 팀원들을 격려해야 한다.

적절한 보상 또는 감사의 표현은 무엇일까? 몇 주 동안 야근을 하는 팀원들에게 팀장이 며칠의 휴가를 요구했지만 정작 높은 관리자는 수고했다며 강제로 술은 권하는 회식으로 마무리 했다고 한다. 이에 팀원들은 적절하지 못한 보상이라며 불만이 오히려 증대된 경우도 있었다.

또한 너무 잦은 칭찬은 역 효과를 불러 일으킬 수도 있다. 실제 스티디 그룹의 K군 같은 경우는, MS의 수석 엔지니어와 협업을 했을 때, 그가 LAN선 하나 설치한 것만으로도 칭찬을 하고, 일일이 Wonderful!, Great!! Awesome!!과 같은 말을 함으로써, 나중엔 '나를 무시하나?'라는 생각까지 들게 되었다고 한다. 우리는 종종 업무 메일에서 자주 접하는 "감사합니다"라는 말을 볼 때 여러분은 감사함을 느끼는가? 그래서 칭찬하는 방법을 알 필요가 있고, 이에 대해 TAPE를 적극 활용할 필요가 있다.

- Things: 눈에 보이는 것을 칭찬하라
- Achievement: 업적을 칭찬하라
- Personal Trait: 성품을 칭찬하라
- Evidence: 구체적인 사례를 들어 칭찬하라

또한 상대방의 개선점을 전달할 때 주의해야 된다. 칭찬에 대한 재미난 실험으로 칭찬을 먼저하고 개선점을 말하는 것과, 개

선점을 말하고 칭찬을 먼저 하는 것에 대해 실험을 했는데, 칭찬(긍정적인 면)을 먼저 말하고, 단점(개선점)을 말하는 방식이 더 상대방에게 긍정적인 생각을 이끌어냈다고 한다.

Innovator

새로운 것을 받아들이기 좋아하는 사람으로 혁신이나 변화를 전파하기 위한 가장 좋은 사람이다. 너무나 새로운 것을 좋아해 오랫동안 나와 함께 있기 힘든 단점이 있지만, 변화를 이끄는 초기에 나의 의견을 동의해 줄 사람으로 꼭 필요한 사람이다. Test the water 패턴으로, Innovator를 Evangelist로 만들기는 힘들지만, 차츰 노력을 통해 변화를 전파하는 사람을 찾아야 하며, 조그만 성공(small success)과 시간을 두고(Time for reflection) 사람을 설득해야 한다.

Innovator를 찾기 위해서 gadget이나 새로운 물건을 즐겨 사는 사람을 찾아보아라. 또는 개발 성향을 볼 때, 새로운 것을 만들기는 좋아하나 유지 보수하는 작업(반복적인 일)에 싫증을 심하게 느끼는 사람을 찾으면 된다. Innovator의 좀 더 자세한 내용은 필자 블로그에 정리한 <http://tinyurl.com/yffhh3a>를 찾아보길 권한다.

Guru on your side

이제 다른 이들에게 나의 생각이 옳다는 것을 알리기 위해 숨겨져 있는 technical leader를 찾아서 조언을 구해 아이디어를 다듬고 포섭해야 한다.

새로운 아이디어의 거부감을 Guru의 권위와 신뢰를 통해 해결할 수 있기 때문이다. Guru에게 조언을 구할 때는 그가 매우 영향력 있는 사람이므로 겸손하게 조언을 구해야 한다. 그리고 조언을 구하기 전 Guru의 특성도 미리 살펴볼 필요가 있다. 너무 비판적으로 바라본다거나 즉각적인 제약사항, 제안사항을 주면서 생각을 묶어 버리는 Guru도 있으니 Connector에게 잘 문의해 나와 코드나 색깔이 맞는 분에게 조언을 구해야 된다. 이러한 조언을 통해 나의 아이디어가 더욱 다듬어 지게 되고 곧 이것이 외부의 검증된 솔루션(External Validation)의 좋은 예가 되기도 한다.

Corporate Angel

이제 Innovator로 쪽수도 채우고, Guru를 통해 권위도 획득했다. 이제 수호천사(고위 경영진)의 힘을 빌려 크면서도 혁신적인 변화를 바로 일으킬 수 있다. 많은 회사가 한 부서에서 변화를 시험하고 성공하면 그 변화를 Corporate Angel의 힘을 빌려 큰 변화를 일으킨다. 또한 변화를 이끄는 사람들을 보상하기 위해

뽀뽀뽀 파와 텔레토비 파 이야기

유치원의 아이들은 자신이 선호하는 프로에 의해 그룹의 성격이 갈린다고 한다. 어떤 그룹은 뽀뽀뽀의 내용을 보고 얘기하는 그룹도 있고, 다른 그룹은 텔레토비에 나왔던 내용을 보고 얘기를 나눈다고 한다. 하지만 만약 두 그룹이 하나의 TV만 봐야 되는 상황이라면, 결국 Big Brother의 힘에 의해 모든 것이 좌지 우지 된다. 이 사람이 나의 편이라면 결정적인 상황에서 큰 힘을 얻을 수 있을 것이다.

윗 사람이 우리를 밀어주고 있다라는 것도 크게 작용한다. Corporate의 힘을 계속 빌리기 위해서는 어떠한 수치고 같은 가시적인 지표로 보여줘야 되고, 변화를 수용하는 입장에서는 그냥 고위 관리자가 말하는 이달의 조언처럼 들을 수 있는 단점도 발생할 수 있다. 그러니 Guru를 같은 편으로 두어 변화의 당위성을 부여하는 것이 중요하다.

Connector를 통해서 Guru를 찾아내고, Guru의 지원을 받아 아이디어를 검증 받고, Innovator를 설득하고 Corporate Angel을 포섭하는 것이 큰 변화를 일으키는 좋은 방법이다. 실제 N사의 경우 유명한 소프트웨어 공학 대가를 포섭해 1년 동안 대가가 하는 방식 그대로 소프트웨어 관리 기법을 하게 함으로써, 아주 수준 높은 품질의 소프트웨어를 얻게 된 사례가 있었다. Guru와 Corporate Angel의 조합의 성공이라고 할 수 있다. 물론 정치적으로 높은 사람이 자신의 의견을 피력하고 싶을 때, 자신의 의견과 부합하는 Guru(컨설턴트)를 데리고 와 팀을 조절하는 경우도 있다.

Small Success

큰 장기적인 목표를 달성하기까지 너무 많은 시간이 걸리기 때문에 중간에 축하하고 팀원들과 성공과 기쁨을 공유하는 것이 중요하다. 프로세스에서 보여지는 진척율도 Corporate Angel에게 중요하지만 팀원들에게는 무언가 더 나이지고 있다고 느낌과 성취감을 주는 것도 중요하다.

스터디 팀원인 cavin님은 다이어트 -25kg 같이 너무 원대한 목표를 세웠기 때문에 실패했다는 경험담을 들려 주었다. 좀 더 주 단위로 적게 목표를 세웠으면 성공하지 않았을까? 이것이 Small Step, Time for reflection 패턴이다. Small Success의 제미난 예로, 풍월당 박 대감님의 남녀 탐구생활의 쇼핑편 이야기(<http://tinyurl.com/ykyfh8g>)를 볼 수 있다. 여성의 쇼핑 전략이 전형적인 Small Success라는 것이다(2분 50초대부터 보시길). 마음에 드는 셔츠를 발견한 후 저것보다 더 예쁜 옷이 없는지 찾아보고, 없으면 결국 마음에 드는 셔츠를 사게 되고, 마음에 드는 셔츠를 샀다는 기쁨(small success) 후에 셔츠에 어울리는

스커트를 찾아 또 백화점을 돌게 된다. 스커트를 사게 되면, 그 기쁨(small success)을 만끽한 채 다른 것을 또 사게 된다. 일종의 small success 패턴이라고 할 수 있다. 단점은 정장 사야 할 물건을 못 사게 되는, 즉 너무 목표를 세분하게 쪼갠 나머지 정작 그 목표에 도달하지 못한 경우다.

이러한 얘기를 나눈 후 프로세스와 팀을 구축하는 것에 비교해서 얘기를 나누었다. milestone이나 scrum과 같은 경우의 개발 방식은 중간 중간 결과물을 공유하고 기쁨을 느낄 수 있기 때문에 small success하기에 아주 유용한 개발 방식이다(하지만 우리나라에서는 이미 테드라인이 정해져 있고, 어느 정도 일정이 연기된 상황에서 시작하기 때문에 Small Success하는 기쁨을 느낄 수 없는 문제점이 있다). 그리고 이러한 프로세스에 맞는 Feature Crew(기능 팀)에 대한 얘기들을 했다. Microsoft 개발자에 Vista를 개발하면서 느낀 절망과 Windows 7을 개발하면서 기쁨을 느낀 글(<http://tinyurl.com/yfdy6hp>)을 보면 이러한 프로세스의 중요성을 실감할 수 있다.

milestone은 자칫하다간 최종 목표가 흐릿해 지는 문제가 있고, 또 반대로 waterfall을 중간에 변경사항이나 피드백을 늦게 받는 단점이 있기에 milestone 방식과 waterfall 방식의 장점을 취한 Staged Delivery 방식도 있다.

Microsoft는 약간 다르게, MQ 라는 것을 도입했다. 하나의 Milestone이 끝나면, 그것을 회고하고 부족한 면과 시장의 흐름이나 동향을 보고, 변경할 사항을 다시 받는 구조를 취하고 있다.

Staged Delivery

Staged Delivery는 소프트웨어 개발의 라이프 사이클(life cycle) 모델이다. 우리는 흔히 Waterfall 방식을 사용했으나 Waterfall 방식은 심각한 문제가 테스트 단계에서 발생되어 유지보수 비용 및 일정 지연 문제가 프로젝트 막바지에 발생하는 경향이 있다. 그래서 Iteration 방식이 출현하게 되었으며 Iteration 방식은 요구분석, 설계, 구현, 테스트의 과정을 Concern 단위로 반복해 진행하는 것으로 내포된 잠재 버그를 최대한 빨리 찾아내고자 했다. 그러나 이 방법 또한 전체 도메인을 이해하지 못하면서 Concern을 나눈다면 위험성이 높은 Concern을 먼저 개발하기 위해 선택하는 일이 힘들어지게 된다. 이를 보완하기 위해 하이브리드 방식인 Staged Delivery 방식을 사용할 수 있는데 Staged Delivery 방식은 분석 및 설계까지는 Waterfall 방식을 사용하고 구현 및 테스트는 Iteration 방식을 사용하는 것이다. 이렇게 하면 요구분석, 설계를 통해 전체 시스템 개발 중 위험성과 중요성이 가장 높은 Concern을 분류할 수 있을 것이고 우선순위가 가장 높게 도출된 Concern부터 Iteration 형태로 개발하는 것이다. Iteration 방법으로 개발을 못한다면 Staged Delivery 방식을 채택하길 권한다.

정리

이번 장은 결국 사람을 만드는 얘기를 했다. 먼저 쪽수를 채우기 위해, Connector를 통해서 Innovator 성격의 사람, Guru를 찾아냅니다. Innovator를 만나 잘 설득하고, 칭찬한 다음 자신의 의견에 동조하는 좋은 동료로 만든다.

그리고 이어서 Guru를 만나, 자신의 의견에 도움을 구하고 Tailor Made와 같이 자신의 의견을 더욱 잘 다듬고, 전체 조직에게 신뢰를 얻어야 한다. 이로써 Innovator로 든든한 홍보세력도 얻게 되고, Guru로 신뢰 세력도 얻게 된다. 그 다음 Corporate Angel에게 도움을 요청해 거대한 변화할 수 있는 전방, 후방의 변화를 다 얻는다.

더불어 Just Say Thanks 패턴을 이용해서 Innovator와 Guru에게 감사함을 표하고, 가능하다면 조그만 정성이 담긴 선물을 함으로써 변화를 주도해 나가면 된다. 다음 기고에서는 좀 더 변화를 활성화시키고 의견을 조직 내에 과급시키는 이야기를 전달하도록 하겠습니다. +



Main IT Contents를 제공하는 Reader's Option.
마소는 독자들이 Success할 수 있는 Opportunity를
드립니다. 보다 우수한 Function을 채택하고자하는
Technical Worker And Real-time Enterprise는
마소에서 해법을 찾을 수 있습니다.

마소의 작은 다짐

시시각각 변하는 세상 모든 얘기를 담을 수는 없지만,
IT에 관해서는 전부를 보여드리겠습니다.
마소 필자와 편집부는 여러분의 호기심을 해결하기 위해
오늘도 최선을 다하고 있습니다.

micro 1년 후에도 내용이 살아있는 집지
Software