エンジニア部門マネージャの役割

篠原 秀

シリコンバレーのエンジニアの多くは，過去の知識や経験を活かしながら，いままでと違った職に移る傾向がある．一部のエンジニアたちの長期的な目的には，「マネージャになる」ということも含まれている．そこで今回は，シリコンバレーにおけるエンジニア部門のマネージャの役割について紹介する．

種類・仕事内容

エンジニア部門のマネージャの役割について新聞やインターネットの企業の求人をみると，

ASIC開発マネージャ
- 7～10年以上の経験
- 数人のエンジニアの管理
- アーキテクチャからシステム立ち上げまでの責任がもてて，ブロック間の依存関係を明確に説明できる能力が必要．

プロトコル開発マネージャ
- 4～5人のプロトコル開発エンジニアの管理．
- 責任範囲は，デザイン，開発，プロトコル・ソフトウェア開発の支援のほか，リソースの確保，計画，予算作成．6年以上の技術経験と1～2年の管理経験が必要．

などの項目が見うけられる．このほか，デザイン・マネージャ，リリース・マネージャ，ソフトウェア・マネージャなど，多様な肩書きや仕事内容が，求人広告には書かれている．

求人での仕事内容はともかく，実際のエンジニア部門で働くマネージャたちの役割には多様性がある．また，会社やポストの責任レベル，個人のスタイルによって，仕事内容は大きく異なる．ここですべてのマネージャたちの仕事について紹介するのは不可能なので，もっとも一般的なマネージャたちの仕事内容のうち，いくつかを紹介する．

チームづくり

チームづくりは，Team Buildingと言われている．新しい部署ができて，最初にマネージャが採用された場合，このマネージャがHiring Manager（採用するマネージャ）として一からグループをつくる．

まず，新しい部署にどのような目的があるのかを把握し，どのような技術者たちを何人採用して，いつまでに製品を作り上げるかなどのプランを立てる．もちろん，このプランの中には，予算が含まれている．人件費やツールなど設備のコストの見積もりをして，場合によっては，ROI[注1]も予測する．採用活動を始めるときには，必要なエンジニアたちの技術レベルやコミュニケーション能力などのイメージをもとに，求人内容を人事部に提出する．

部署の目的が同じでも，マネージャによって採用する人材の条件には，かなりの差があるようだ．あるマネージャは，一見バラバラの経験をもったエンジニアたちを採用しながらも，技術メンバの弱点をそれぞれ補うチームづくりを実現させる．対照的に，粒がそろったメンバを採用して，チームに一貫性をもたせながら生産性をあげているマネージャもいる．

チームづくりで重要なことは，チームにスケーラビリティをもたせることである．チームをゼロから起こすとき，最初の数人は，マネージャ自身が慎重に選択する．この段階が終わると，採用されたエンジニアたちが，自分と同じレベルのエンジニアたちや経験や能力の補助をし合えるエンジニアたちの採用を推薦するようになる．またチームがある程度成長してくると，経験の少ないエンジニアたちをサポートしながら仕事をこなしたり，マネージャの細かい指示なしで開発方法を改善していったりする．

マネージャがあまり一人でがんばりすぎると，スケーラブルでないグループができてしまうときがある．エンジニア時代にやり手だったマネージャにこの傾向があるが，どちらかというと一人でなんでもやってしまうタイプである．彼/彼女らは，過去の成果を買われてマネージャに昇進するわけだが，グループの生産性がよくならないので，余計に一人でがんばってしまう．結果的には悪循環をつくってしまうわけである．どうやら，エンジニア時代では，「自分

注1：Return on Investment．はやく言えば出費 vs 収入のこと．普通Profit Lossが計算される部署の責任者は，重役クラスのマネージャが多い．

〔表1〕エンジニアとマネージャの対処の違いの例

イベント	エンジニア	マネージャ
ツールや設備などの不具合	支援を頼む．自分で解決．	問題解決のための解析や支援のコーディネート．
生産性の低下	残業をする．	原因究明と対策． 人員を増やす．支援を頼む．方法論を向上するなど．
会社への悪いニュース	うわさを聞く．新しい仕事を探す．	事実を確かめ，エンジニアたちへ説明をする．なだめる．
査定時期	昇給などの要求．	人事部や上司へ昇給の推薦． 予算，生産性，結果などを基に昇給幅をエンジニアたちに説明．
エンジニア採用	面接また推薦．	推薦などを元にグループやプロジェクトへの影響を考える．
機能の追加	説明を聞いたり，異議を唱える．	リスクを報告したり，追加の理由を開発グループに説明． スケジュールの変更．
技術的な問題	新しい方法を導入． 知識や情報を同僚やマネージャから求める． 自分で解決．	問題点を把握して，アドバイスを行ったり，適した人材を探す．

を磨いて成長する」をモットーとしていても，マネージャになると「人を成長させる」ということに変えなければならないようだ(**表1**)．

チームの生産性をあげる

開発チームを創ったり，増員したマネージャたちは，チームの開発能力を伸ばすために苦心する．とくに新しくできたチームは，チームメートとのやりとりも確立できていないうえ，開発チームとしての完成度が未熟の場合が多い．チーム内での決まりごとや方法論，コミュニケーションの仕方など，最適化しなければならないことがたくさんある(下掲の**コラム**を参照)．

筆者の過去の経験であるが，納品期限に追われて毎晩残業を重ねていた会社があった．そこでは製品開発をすることのみが重要視されていたため，チームの成長がおろそかになっていた．毎晩の残業で，ある程度は開発期間を短縮することができたのだが，納品後のエンジニア部門の能力(efficiency「能率」とeffectiveness「効果」)はひどいものになってしまった．

エンジニア部門を「牛」，製品を「ミルク」にたとえてみよう．管理者たちが，納品期限のみを重視して短絡的なスケジュールをつくってしまうと，牛のミルクをできるだけ早く，しかも多くとろうとする．スケジュールが遅れてくると，ミルクを搾る時間を増やす．がんばった結果，最初の製品はなんとか納品できるだろうが，牛はやせ細ってしまい，病気がちになってしまうだろう．

これをチームづくりで考えた場合，ミルクをたくさんとるためにこの牛を育てていくことになる．管理者たちは，頻繁に牛に運動をさせたり，よいえさを与える．ミルクの質，量，搾乳時間は，牛の世話をした成果としてあらわれてくる．

最近は，開発チームの集団採用というかたちで生産能力の高い企業の買収も頻繁になっていて，ミルクの価値よりも牛の価値に着目している会社も少なくない．シリコンバレーの開発環境では，この「ミルク搾り」と「牛育て」のハイブリッドなかたちがよくみられる．

マネージャたちが行う，チーム作りの仕事の一部には，保守的な要素もある．たとえば，会社の調子がおかしくなったり，チームの意欲が落ちないように，エンジニアたちを精神的に支える仕事もある．会社の状態が良いとき，悪いときを問わず，頻繁にグループや一人一人のエンジニアと会話をして常時コミュニケーションに気を配るマネージャたちもいる．また一般的に，シリコンバレーのエンジニアたちは雑用に時間をとられることが少ない．技術的な知識が必要な雑用は，デザイン・マネージャ[注2]などが，開発の合間にこなしている．

注2：Design Managerは，自らも開発チームに加わりながらもグループをまとめていくハンズオン(Hands-on)マネージャのことで，根はエンジニアという人も多い．一つのプロダクトやプロダクト・ファミリの支援に従事する．チームをまとめたり，ほかのグループとのコーディネートなどを行う．またアーキテクチャ，デザイン，検証などの工程にも参加する．デザイン・チームが書いた記述のチェックや検証チームのテスト・プラン作成を支援したりもする．

コラム　Capability Maturity Model(能力成熟度モデル)

これはカーネギーメロン大学のソフトウェア・エンジニアリング研究所(SEI；The Software Engineering Institute)がソフトウェア開発を促進するために開発した仕様で，略してCMMという．

かつては，ハードウェア開発で成功した開発プロセスもソフトウェア開発では役にたたない場合が多かった．元来，ソフトウェア開発の効果を上げるためにつくられたCMMもハードウェア開発の複雑化に伴い，ハードウェアの開発者がこの仕様に基づいた開発プロセスが必要になっている．この仕様によると成熟度は，5レベルに分類される．

(1)初期レベル
(2)反復できるレベル
(3)定義されたレベル
(4)管理されたレベル
(5)最適化するレベル

開発グループができたての頃は，レベル1の状態である．どのようにしてレベルをあげていくかという仕様も同じくSEIから提供されている．

成熟度をあげるには，プロセスの確立などを要するのだが，各レベルを上げるのに相当時間がかかる．シリコンバレーの半導体開発に属するエンジニアリング・マネージャのなかでCMMを故意に活用している例は見かけないが(知らない人も多い)，成果をあげている開発グループのマネージャたちは，結果的にCMMに沿ったプロセスやプロジェクトの管理を行っている．

http://www.sei.cmu.edu/cmm/cmms/cmms.html
(日本語版は，http://www.iijnet.or.jp/sea/CMM/)参照．