

リーダーを困らせる厄介な部下

チームのリーダーにとって、扱いの難しい部下に振り回され、チームの生産性やモチベーションが低下するのは避けたいもの。部下の特性を理解し、チームのパフォーマンスを上げるための対処法を解説する。



加藤 大介 オイコス メンター

とある居酒屋の奥まった席から、同じ会社のチームリーダークラスと思しき4人組の会話が聞こえてきた。ひとしきり経営に対する不満やそれぞれの上司の愚痴を並べた後に、今度は部下の話題になった。

Aさん「そういえばさ、うちのチームのS君なんだけど、全然言うこと

を聞かないんだよね」

Bさん「反発してくるの？」

Aさん「反発というか、まるで報連相をしないんだ。こちらからいちいちどうなっているか聞かないと、何も言ってこない。仕事はきちんとこなすやつなんだけど」

Cさん「その人、すごく優秀なんじゃないの？期限と成果だけ示せば必ず

完遂するタイプ。困ったことがあっても多少のことはなんとか解決しちゃうので、何かあって報告に来るときはホントに重大なときだけっていうタイプ」

Aさん「そうかもしれない。いつか大きな問題が起こりそうで、すごいストレスだよ」

Dさん「性格は矯正できないしね。でも皆にもいろんな部下がいるでしょ。手こずる部下とかいない？」

Cさん「実はうちにも仕事にムラがある問題児がいる。自分が理解している部分は熱心に詳細な作り込みをするんだけど、逆にあまり理解できていない部分は手を抜いているような感じなんだ」

Aさん「質問しに来ないの？」
Cさん「進捗報告も定期的にしてくれるし、完成したら『レビューをお願いします』って自主的に来るし、それはいいことなんだけど」

Dさん「さっきの報連相しないやつ

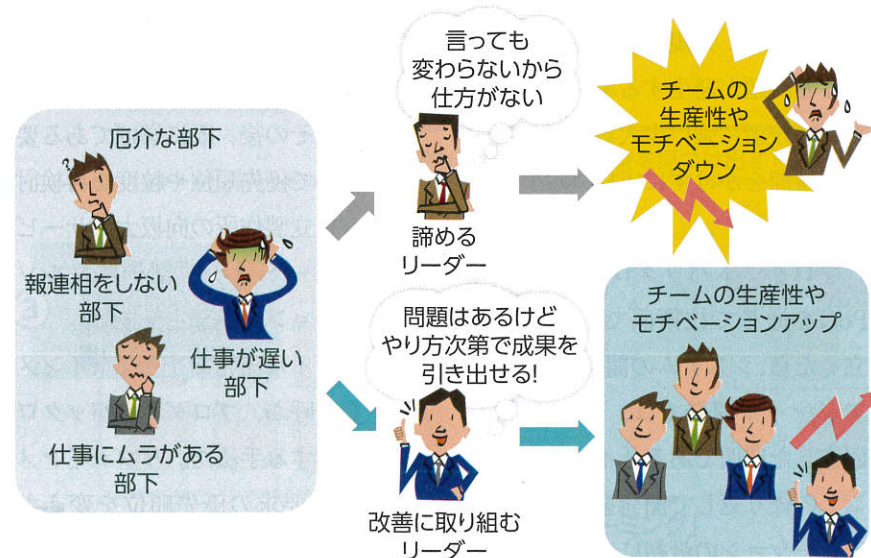


図1 リーダーは厄介な部下からも最大限のパフォーマンスを引き出す必要がある

より、よっぽどましじゃないか」
Cさん「ある意味ではましかもね。でもダメ。レビューすると明らかに適当な仕事をしているのが分かったやうだよ」

Bさん「でも彼は全然仕事ができないわけじゃないよね。うちにはホントに仕事が遅いやつがいてね。作業は遅いし段取りも悪い。だからいつも慌てている」

Dさん「うちにもいそうだな」

Bさん「それだけじゃないんだよ。仕事が遅いのにも全部自分で抱えちゃって、その尻拭いがもう大変で」
Dさん「いやな上司も問題だけど、部下にも悩まされるね」
一同「全くだ…」

この居酒屋での会話に出てきたような部下、あなたの周りにもいないだろうか。職場には様々なメンバーが存在する。仕事ができるメンバーもいれば、そうでないメンバーもいるだろう。厄介な部下への対処に頭を悩ませているリーダーは少なくないはずだ。

居酒屋で話題になったのは、以下の3つのタイプの部下である。

- (1) 報連相をしない
- (2) 仕事にムラがある
- (3) 仕事が遅い

このような部下に振り回され、チームの生産性やメンバーのモチベーションが低下することは避けたい(図1)。リーダー自身も、思うようにいかない部下に気を病んで体調を崩したりしては大変だ。

冒頭の居酒屋の場面のよう、悩

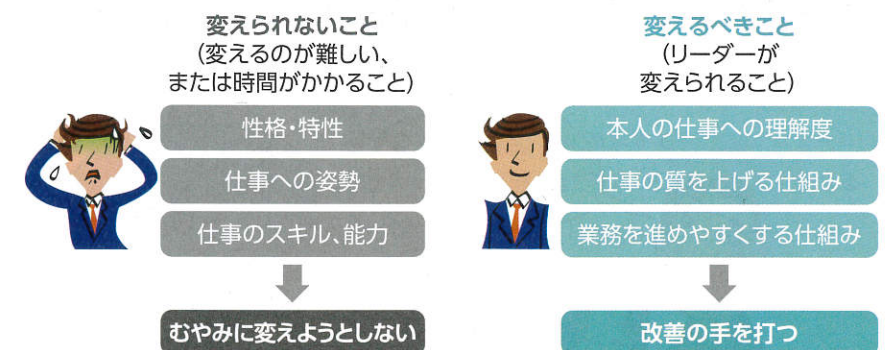


図2 「変えられないこと」と「変えるべきこと」を分けて対処

みを言い合っているだけでは解決しない。また、「誰だ、こんなやつをチームにアサインしたのは」とか「あの性格をなんとかたたき直さねば」といった発言をよく耳にするが、気持ちでは理解できても怒ってみたところで意味はない。問題の部下は、既にチームの一員としてアサインされているのだ。

もちろん、「何を言っても変わらないから仕方がない」と諦めてしまうのはリーダー失格である。リーダーは、たとえ厄介な部下を抱えてその仕事ぶりに悩まされたとしても、でき得る改善の手を打ち、部下から最大限のパフォーマンスを引き出す必要がある。

変えられないことと変えるべきこと

そのときに必要なのは、部下の特性や行動を理解し、「変えられないこと」と「変えるべきこと」をきちんと分けて対処することだ(図2)。リーダー自身が、思うようにいかない部下に気を病んで体調を崩したりしては大変だ。

例えば、部下の性格や特性は、そう簡単に変えることはできない。仕

事への姿勢も同様だ。例えば前出のように、自分の理解していることはしっかりやる、といった仕事のムラは、本人の仕事に対する姿勢や習慣に問題がある。しかしこれは、口で言ってもすぐには変わらないことが多い。仕事をこなすスキルや能力については、経験によって向上が見込めるとしても、時間がかかる。

一方、変えるべきなのは、メンバーの仕事の進め方やメンバーへのアプローチの仕方だ。仕事への姿勢は変わりにくいとしても、その仕事の目的や内容についての理解度を上げることは可能だ。それによって、部下の仕事の進め方を変えられる。周りのチェックによって仕事の質を上げる仕組みや、部下が業務を進めやすくする仕組みを取り入れることも、メンバーの仕事の質を高めるために有効だ。例えば、成果物のレビュー方法を変えるのはその一例である。

リーダー次第で、メンバーの仕事ぶりは変えられる。次ページからは、前述した3つのタイプの厄介な部下について、仕事のパフォーマンスを上げるためにリーダーが取るべき対処法を解説していく。

報連相をしない部下 報告しやすい環境は「おひたし」



流通業系SI企業で決済システムの刷新プロジェクトのプロジェクトマネージャーを務める谷村は、チームメンバーの石田にデータ構造設計を担当させていた。石田には、毎週どこか作業のきりがいいところで作業報告をするように伝えてあった。

ところが石田は谷村に報告をほとんどしてこなかった。そしてある日、石田がおずおずと谷村に報告したいことがある、と言ってきた。ユーザーからのヒアリング内容に不備があっ

たことに気付かず作業を進めてしまったため、再ヒアリングと現状調査が必要になった。現状ではとても期日に間に合いそうにないという。

谷村は、いまさら何を言うか、と石田への怒りを募らせた。

読者の皆さんの中には、部下に対して「報連相」の徹底を口うるさく指導している人は多いのではないだろうか。部下であるメンバーの側も、新入社員のときから「報連相はこまめにせよ」と耳にタコができるほど聞かされているはずだ。ところが現実には、報連相をきちんとするタイプと、いくら言っても報連相ができない、あるいはしないタイプとの、歴然と差がある2者が存在する。

機転が利くことを求めない

報連相ができる部下は、リーダーが何を求めているのかを理解してい

る。報告の目的を理解し、伝え方、タイミング、報告の粒度などが的を射ている。いわば、「機転が利く」と言ってもよい。逆に報連相ができない部下は、そもそも報告の目的が曖昧、または全く理解できていないことが多い。そのため聞かれたことしか話さない、聞かれないと報告しない、表面的なことしか報告しない、などということになる。

機転が利く部下なら自分で判断し、考えて適切な報連相ができるだろう。しかし全てのメンバーが機転が利くわけではないし、機転を利かせることを求めるのも間違いだ。機転の利かせ方は、経験を積んで身につく人もいれば、身につかない人もいる。そこを改善することに力を入れるのは無駄といえる。

機転を利かせることを求めるのではなく、メンバーが自発的に適切な報告をする状況を作る必要がある。そのためにリーダーが変えなければいけないのは、部下に対するアプローチだ(図3)。対処策は4つある。

(1) 根気強く目的と内容を伝える

報連相ができない部下は、報告とは「経過や結果についてこまめに知らせること」程度の認識しかない。リーダーがなぜその報告を求めているのかを理解していないのだ。目的が分からなければ、報告すべき内容も曖昧なままだ。結果として、「仕事はちゃんと進めているんだから、いちいち報告しなくてもよい」といった行動を招いてしまう。

こうした行動を変えるには、まず

は報告の目的を部下ときちんと共有することだ。例えば週次の作業報告であれば、「ユーザーヒアリングの抜け漏れがないか洗い出すため」「進捗状況に応じて後工程の要員を調整するため」といった目的を共有する。それによって、部下は自分の仕事に関わるどんな情報を報告すべきなのかを理解できる。

その上で、「誰に」「どのようなタイミングで」「何を」「どのくらいの粒度で」「どのような手段で」など、細かく定義することも検討したい。そこまで必要なのか、と思うかもしれないが、できない部下にやらせるためには、指示内容は細かく定義しておくことが望ましい。

(2) 報連相しやすい環境を作る

報連相をしやすい環境作りもリーダーの仕事のうちだ。最近では部下の報連相を「『おひたし』にして返す」と言うことがある。おひたしはほうれん草などの調理方法だが、ここでは「お:怒らない」「ひ:否定しない」「た:助ける」「し:指示する」の4つを意味する(図4)。

報告するたびにリーダーが嫌な顔をする、というのは部下が報連相をしなくなる要因の1つである。

悪い報告をしなければならないとき、報連相ができる部下は、自分がその場で怒られることについては仕方がないと割り切って、それよりもリーダーに一報を入れることを重要だと認識できている。しかし、怒られるくらいならば様子を見ようと思う部下もいる。それが手遅れ

お 怒らない
ひ 否定しない
た 助ける
し 指示する

図4 部下からの報連相に対する上司の心構えは「おひたし」

- × 悪い報告のときに部下の不手際を怒る
- 悪い報告こそよく報告してくれたと評価する
- × 部下の意見を頭ごなしに否定する
- いったん部下の意見を認めてから、反対意見を伝える
- × 対策を自分で考えて実行するように突き放す
- 問題への対策を共に考え、適切な指示を出す

になるような事態を引き起こす。

リーダーは悪い報告のときに報告者の不手際を怒るのではなく、むしろ悪い報告こそよく言ってくれたと評価すべきだ。その上で、問題の対策などを部下と共に考えることで部下を助ける。そして適切な指示を出すことが大切だ。

否定しないのも目的は同じだ。たとえ部下の意見や提案が間違っていると思っても、頭ごなしに否定するのではなく、いったん部下の意見を認めてから、なぜ間違っていると思うのかの理由と共に反対意見を伝える必要がある。

部下が何か話しかけてきたときに、きちんと部下の方を向いて対応することも大事だ。忙しいからといってディスプレイを見たままキーボードをたたきながら話を聞いていると、部下は自分の存在を否定されていると感じてしまうので要注意だ。

(3) 自分の情報を共有する

大切なことは、リーダーからも報連相を行い、自分が持っている情報を部下と共有する姿勢を見せることだ。自分に報告してほしい情報が何

のために必要なのか、他のタスクとどのような関連があるかなど、報告事項の意義を部下に理解させるために必要な情報である。部下の作業には直接関係しないようなことでも、関連する情報などは積極的に共有するとよい。互いに情報共有し合う機会を増やすことで、部下は自分から進んで報連相をするようになる。

(4) 相手の考えを引き出す

部下からの報連相に対してアドバイスをするときは、つい手取り足取り、細かな指示を出してしまいがちだ。これでは部下が自主的に行動する習慣が身につかず、言われたことしかやらないようになってしまう。

そこで、アドバイスの際には部下の考えを導き出す工夫をしたい。「君ならどうする?」「こういう考え方とこういう考え方があるけど、どっちが適切かなあ?」などとヒントと質問を繰り返す。部下に考えさせ、正しい答えを導き出させるのだ。相談という形を取った育成で、コーチングにもつながる手法だ。部下は「自分で答えを導き出した」という実感を得ることができ、モチベーションの向上につながる。

(1) 根気強く報連相の目的と内容を伝える

(2) 「おひたし」で報告しやすい環境を作る

(3) 自分が持っている情報を共有する

(4) アドバイス時は部下の考えを引き出す

図3 報連相ができない部下への対処策



仕事にムラがある部下 中間レビューで作業品質高める



保険会社の自動車保険新システム構築プロジェクトにおいて、プロジェクトリーダーの長尾はメンバーの松本に、Web機能と連携したサーバー側の機能の設計を任せた。松本はメインフレームでの経験は豊富だが、Webシステムの設計・開発は初めての挑戦だった。

ある日、長尾が松本の成果物の途中経過を確認しようとしたときに、問題が判明した。松本の設計は、彼が自分で理解できている箇所は詳細に作り込まれているのだが、ところどころ文字を埋めただけのような箇所が散見される。どうやら松本があまり理解できていない部分のようだ。松本に聞くと「レビューやテストのときに確認するつもりだった」という返事が返ってきた。

自分が理解しているところや興味があるところは徹底して作り込む

が、よく分からないところは適当に済ませてしまう部下がいる。いわば、自分の好き嫌いがそのまま仕事の成果に影響してしまう、仕事にムラがあるタイプである。リーダーにとっては、安心して仕事を任せられない、厄介な部下といえる。

いい加減に作ることを容認

このような部下に対して、どうすれば興味のない仕事や理解できていない仕事であっても、こちらの意図通りに成果を出させられるようになるだろうか。

部下の仕事にムラがある場合、本人の性格や特性を取り上げても解決には至らない。興味のある仕事や自分が理解している仕事にばかり力を入れてしまうのは、本人の特性といえる。これは、簡単に変わるものではない。

この場合、真っ先に変えるべきも

のは、仕事の手順や仕組みである。具体的には、仕事の成果をチェックする工程を増やすことが、対策の1つになる(図5)。仕事のムラにつながる作業上の問題を早期にあげり出せるようになるからだ。

(1) 中間レビューで問題を早期発見
冒頭のケースで、松本さんの「レビューやテストのときに確認するつもりだった」という発言は、仕事にムラがある部下の典型的なパターンだ。彼らに「レビューとは何か」と質問すると、「確認することだ」という返事が返ってくることが多い。自分が作った成果物が正しいかどうかを確認する場という認識である。

この認識は間違いとはいえないが、レビューを受ける側の仕事に対する姿勢に安易な気持ちを生じさせる場合があるので注意したい。つまり、「レビュー＝成果物確認」という認識は、「分からないところはレビューのときに確認すればいい」と考えていることの表れでもあるからだ。「いい加減に作る」ことを容認しているようなものだ。

実際、そういう認識のメンバーがレビューを受けると、その場で仕様確認などの質問がよく出てくる。不明点は作業している段階でその都度質問せよ、と徹底しても、なかなかうまくいかないことが多い。

筆者も、まだ経験2～3年目だった頃は、レビューは自分の成果物を確認、チェックしてもらう作業だと思っていた。そのため「これはこのように解釈して作ったが合っている

か」など、質問会のようなレビューを繰り返していた。レビュー者の中には、「そんなことを今さらチェックさせるのか」と怒る人もいた。今思えばとても恥ずかしいことだ。

問題なのは、成果物が出来上がるまで、何も確認しないことだ。成果物の方向性や品質に問題があることがレビューで判明したら、多大な手戻りとなってしまう。

こうした状況を避けるためにも、中間レビューを取り入れるとよい。中間レビューは、成果物が完成する前の段階で、作業の方向性や作業量などについて確認するレビューである。中間レビューのチェックが前提となれば、興味のない仕事や理解できていない仕事であっても、手を抜くことは難しくなる。

中間レビューのやり方は作業や成果物の規模によって変えればよいが、ここでは3段階の中間レビューを紹介しよう(図6)。

第1段階は、作業の構想が30%程度まとまった段階で、方向性を確認することだ。第2段階は、作業が50%進捗した段階で、残りの作業のボリューム感を確認するレビューである。そのタスクの残り半分を仕上げるための時間を把握し、当初の期限の妥当性も検証できる。

第3段階の中間レビューは、作業の進捗が80%程度のときに実施する。ここでの主な目的は、成果物品質の確認である。成果物の粒度や作りは問題ないか、システムの目的に合致したものになっているのかなどを確認する。また作り込みの程度の

すり合わせを行うことも大切だ。

(2) 最終レビューは正当性証明

成果物品質を上げるためには、中間レビューの実施に加えて、最終レビューに対する意識改革も実践したい。前述の通り、メンバーが「レビュー＝成果物確認」という認識でいると、成果物の完成度が低くなる要因になり得る。この考えを改め、レビューの位置付けを見直すべきである。

筆者がチームリーダーとしてレビューする側になったとき、「レビュー＝成果物確認」という認識が正しいのか、疑問に感じるようになった。そして、レビューを効率的に、手戻りが少なく、高品質の成果物を生み出すプロセスにしたい、と考えて出した結論は、「レビュー＝レビュー者に対して成果物の正しさを証明する」ということだった。

レビューを受ける側は、自分の成果物の正当性を証明する。そのために前述の中間レビューにて方向性、ボリューム、品質を確認し、最終レビューにて成果物の正当性をレビュー者に証明する、これをレビューの定義としたのだ。

メンバーには「私に君たちの成果物の正当性を証明できるよう、完成品を持ってくること。正当性を証明するために必要な資料は第三者でも理解できるように整理しておくこと」と指示するようにした。

当初は戸惑いを隠せなかったメンバーだったが、3段階の中間レビューで方向性、ボリューム、品質の度合

中間レビューを実施して
(1) チェックポイントを設け、
問題を早期にあげり出す

最終レビューを「確認の場」ではなく、
(2) 成果物の「正当性を証明する場」と
位置付け、意識ややり方を変える

図5 仕事にムラがある部下への対処策

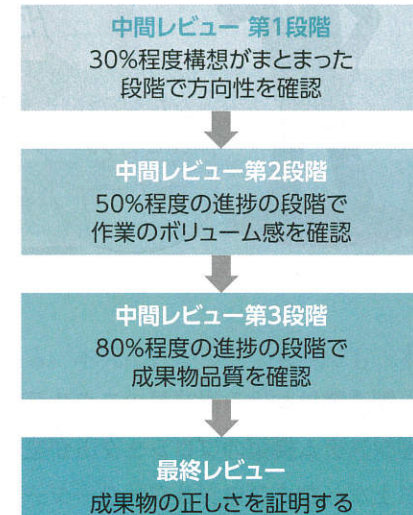


図6 中間レビューでチェックする

いを確認し、疑問点も解消でき、レビューでは最終成果物の正当性の証明を行うことで、レビューによる手戻り作業は激減した。

分かりやすく説明する必要性があるので、ソースコードにしてもドキュメント類にしても、レビュー時点で読みやすく分かりやすい、内容も充実した高品質な成果物が作られるようになった。

手順や仕組みを変えるのは、面倒で手間がかかることだ。しかしそれによって部下の成果物品質のムラがなくなれば、余計な手戻りやイライラ感からは解放される。ぜひ取り組んでいただきたい。

仕事が遅い部下 作業の手順書を一緒に作る



製造業における従業員のスキル管理システム構築のプロジェクトが立ち上がった。プロジェクトリーダーの山本は顧客要求の取りまとめを、新しくチームに加わった森に任せた。森は以前、製造業の人事系システムの開発に長く関わってきた経験がある。その森を山本がプロジェクトに引き入れたのだ。山本は、真面目な森の態度や話しぶり、関連知識の豊富さを評価していた。森の過去の成果物は分かりやすく、資料作成能力の高さにも期待していた。

プロジェクトが始まってしばらくすると、山本は森の仕事ぶりに疑問を感じるようになった。森は毎晩遅くまでデスクに向かっており、時折休日出勤もしている。他のメンバーの多くは残業は少なく、定時で退社している者もいる。

森には、手が空いているメンバーに作業を依頼するように伝えている。

しかし、山本がそれとなくメンバーに聞いてみると、森からの指示は大雑把で、成果物が何に使われるかも不明なものが多いという。結局、話を聞いているうちに依頼が棚上げになってばかりとのことだった。

週次の進捗報告を重ねるうちに、進捗が思わしくないことが分かってきた。山本が追求すると、森は同じような作業を何度もやり直していることが分かった。手戻りが多発しているのだ。

その後、作業の進め方について何度説明させても改善せず、森は「一生懸命やっています、進捗の遅れを発生させないように頑張っています」と繰り返すのみだった。

これは、仕事が遅く、作業を抱え込んでしまう部下のケースである。チームで作業を進めるシステム開発プロジェクトにおいて、1人の部下

の作業が遅れると、チームメンバーにも多大な迷惑がかかる。リーダーは、メンバーの特性をよく考慮して、仕事の任せ方や介入の仕方を工夫する必要がある。

経験を見込まれてプロジェクトに入ったはずの森さんだが、山本さんの期待に反してその仕事ぶりは到底満足のいくものではなかった。夜遅くまで仕事をしていても作業は終わらず、周りのメンバーに作業を依頼することもなく仕事を抱え込む。

実は、森さんが以前のプロジェクトで評価の高い成果物を生み出していたのは、ユーザーやリーダーからの的確かつ詳細な指示があったからだった。

工夫が苦手で段取りが悪い

森さんのように、手順や段取りを的確に指示されれば精緻に仕事をこなす能力を発揮する一方で、自分で考えたり工夫したりしての作業は苦手という部下がいる。やるべき作業が細かく指示されていないと、途端に仕事の進みが遅くなる。

仕事が遅い原因の1つに、段取りの悪さがある。システム開発においては、複数のタスクを1人のITエンジニアが担当するのは当たり前のことだ。例えば要件定義の工程なら、ユーザーのヒアリング、要求の分析、業務フローの作成など、様々な作業をこなす必要がある。

複数あるタスクを整理して、作業の優先順位付けやスケジュール配分などを考慮し、自分の作業の段取りを決める必要がある。しかしこの段

取りが悪くと、仕事がなかなか終わらず、手戻りが発生してさらに遅れるといった事態を招く。さらに、段取りができなければ、周りのメンバーに仕事を依頼することもままならない。結局、作業を自分一人で抱え込んでしまい、事態を悪化させている(図7)。

段取りが悪く、仕事を抱え込んでしまうタイプの部下に対して頭ごなしに「どうしてそんなやり方で進めるのだ」と問い詰めても、そう簡単に仕事の進め方が変わるわけではない。若手のうちならば教育や経験によって段取りのスキル向上が見込めるが、ある程度のベテランにも、このタイプはいる。仕事の進め方は経験や性格によるところが大きい。経験を積んだベテランほど、仕事の進め方を変えさせることは難しい。

作業を抱え込んで常に締め切り直前の状態になっているタイプのメンバーには、自分で工夫させたり、考えて行動させたりすることは有効ではない。「作業に没頭する＝タスクをこなしている」という図式が頭の中で出来上がっている場合が多いからだ。彼らの多くは、「自分は仕事が遅い」という自覚がない。むしろ自分なりに一生懸命やっていると意識が強く、毎日夜中まで頑張っていることを評価してほしいというくらいに思っていることが多い。

このタイプの部下は、適切な手順書などで正しく作業(オペレーション)することの方が得意だ。その方がパフォーマンスが上がり、本人の自信にもつながる。



図7 仕事が遅い部下の問題点

(1) 細かな作業内容を一緒に作る

そこで、取るべき対処策は、指示の出し方を変えて、業務を進めやすくしていくことだ(図8)。

具体的には、作業の目的と到達点を理解させた上で、細かな作業指示までするしかない。さらに、担当業務のWBS(Work Breakdown Structure)を作成させ、ワークパッケージごとに目的や作業内容、関連するスケジュール、品質要件など、いわゆるWBS辞書を整備させる。同時に、ワークパッケージ内の作業手順なども別途作成する。

ただし、これらを何の指針もなしに一人で作成させると、いつまでも終わらないことは目に見えている。対話し、相手に考えさせながら作成することがポイントだ。その手順書に従って作業できるように作業指示を出す。作業の中で何か独自にやらせるのではなく、「作業の入り口で抑え込む」策が有効だ。

(2) 理解できているか説明させる

また、このように仕事を抱え込んでしまうタイプには「仕事に必要な様々なことを実はよく理解できていない」という事象が共通して見られ

(1) 細かな作業内容や手順書を一緒に作成する

(2) 仕事の内容を説明させて理解しているか確かめる

図8 仕事が遅い部下への対処策

る。本人は分かったつもりでも、本当は分かっていないから他者との作業分担もできないし、先々を見通せないから段取りも悪くなる。

理解できているかどうかは、WBS辞書や手順書を作成する際に本人に説明させることで確認するとよい。「このプロジェクトの目的は?」「あなたの役割は?」「ユーザーの期待は?」「あなたの成果物は?」「今あなたがすべきことは?」など、いろいろな切り口で説明させよう。

自分の口で語らせることで、本人が何を理解できていて、何を理解できていないかが明確になる。理解できていない箇所は、その場で理解させるようにしたい。



加藤 大介 (かとう だいすけ)

独立系 Sier やシンクタンク系 IT 企業にてプロポーザルから運用設計まで、数多くのシステム開発プロジェクトに従事。その後、経営企画部門にて人材開発業務(社員研修企画、新卒採用など)の責任者を務めた。現在はオイコスのメンターとして研修講師のほか、公認会計士事務所にて関与先への IT コンサルティングも行っている。