

TPiCS レポート

「これは面白い話ですね。次回のレポートで取り上げさせて下さい」

しばらくぶりにご来社下さったユーザーさんとの会話です。

随分前からTPiCS-Brainを使って頂いているはずなのですが、途中ご担当が変わったりして、これまであまりかんばしくありませんでした。それが去年の夏研修会にご参加頂き、この半年ほどようやくスパートが上がり、最近やっと動き始めて来ましたとのこと。

「しかし導入を決定した当時と比べ、生産量が3倍に増え、機種数もかなり増えてきて、この製品はあの時、親会社にとっても新規事業だったのです。

仕事が増えたのはいいんですが、実は最近、二ノ宮さんがよく言うように、親会社から短サイクル化をしきりに言われるようになりまして、困っているんです。

今、月に一度の発注をしているのですが、月に2度、3度の手配に変えろと言う訳です」

「部品の共通度はあるんですって？」

「ええ、親会社が共通化にも力を入れ最近かなり多いんです」

「ロットまとめなんかはどうですか？」

「そう、当時は注文のあった分だけしか作らないようにしていたんですが、同じようなものが繰返しきますからね。

現場が、来月を横目で見ても勝手にまとめて作ることも有るようです。

親会社からは日別機種別の計画が来まして、それを束ねて前の月の内に作るようにしているのですが、それがコロコロ変わるんです。

共通部品が多いですから、全部締めるとそう大した変動でもないんですが、うちは内示の段階で月単位の手配の製番で走っていますから、それを替えるのが大変なんです。

計画を変えるのが大変だから、前の計画はそのままにして、追加だけの処理で済ますことが結構あるんです。

そうすると、御存知の通り何時か在庫の山です。先日棚卸をしたら、半年分以上の在庫の物がありました」

「そうですか それでは、この半年の苦労は水の泡ですね、このままTPiCS-Brainを使っても巧く行かないでしょう。残念でした」(笑)

随分前からのユーザーさんですから、冗談を言ってからかいます。

「一日も早くTPiCS-IIIかIVに変えて下さい」

「だって、やっとここまで来てと言うか、

まだまともに動いてないのにと言うか、今“システムを変えます。”なんて言えないですよ！」

「だって、やっても大した効果がないんだから やったってしょうがないでしょう」

今回のテーマ

- TPiCS-IV について
- 実績と進捗と在庫について
- ハードウェアの障害について



古来 人間の頭の数だけある“失敗”の、最も大きな原因の一つを、ここでも見る思いがします。

全ての物がトータルで、全ての物が変わりゆくのに、自分だけは変れない、本当は自分も変ることから逃れられる訳ではないのだが、その変化を受入れることができない。

最近 こんなことをよく考えるのですが、まさにこの件も、それにピッタリの事例のように思えます。

世の中が、これほど大きく変わっていて、自分だけ変らずにいられるはずがないのですが、“考え方”だけはな

かなか変えられないものです。

やってみれば、“首をちょっとひねる”のと同じくらい簡単な変化でも、“考え”や“心”を変えると云うのは、なかなか大変です。

逆に変わってはいけないうような事になると、どういう訳か、簡単に、変ってしまいます。

うまく行かないものです。

はたで見ていると、

「もう少しこだわりをなくせば、もっと楽になるのかな！」

「ちょっと 考え方を換えれば、....」
 というようなことが、よくあります。
 ご本人は 至極真面目に考えているのです、
 「いま、... しているから...」
 「... との関係が...」
 「... も 管理したいから...」

私のような立場の人間から見ると 歯痒い思いをいたします。

このユーザーさんの場合、今 思い返しても当時は、T P i C S - B r a i n で良かったのです。

- ① 受注をしてからの手配で 間に合う状況であり、
- ② ロットまとめもしない、
- ③ わりと社内の工程が 深く、手配面より、社内の工

- 程の管理が、狙いの中心でした。
- ④ 受注注番毎の追いかげや、
- ⑤ 作業工数の実績管理、原価管理
等を、目的にしている、
それが、ただ 世の中の動きにつれ、変わっただけなの
です。

「T P i C S を使っていて良かったですね！これが、
 オフコンで 手作りシステムを開発していたら、出来
 上がったころには、“浦島太郎”ですよ。
 違うシステムに変わるのもしんどいでしょうが、しん
 どいだけで済むんだからよかったですよ。
 手作りでは、たとえ思った通りのシステムができた
 としても、“思う”ことが違って来るんですから ど
 うしようもないですね」

T P i C S - IV について

T P i C S - IV のマニュアルの印刷が、今週中に完了いた
 します。

もうしばらくお待ち下さい。
 IV になって、発注注番で実績が管理出来るようになり、
 運営が大分楽になったようです。
 III と比べ、自由度が少なくなった訳ですが、かえって
 その方が楽に運営できるようです。

III を長く使って頂いているユーザーさんから、「IV にな
 って、補助輪が一つ付いたようですね」と言われま
 した。
 “言い得て妙”です。

実績と進捗と在庫について

前回のレポートで 尻切れトンボになっていた このテ
 ーマを、今回 もう少し続けさせていただきます。

実績が入ると、計画に対する遅れ進みが必ず発生します。
 遅れ進みを、次の所要量計算にどう繋げていくか？と
 いうのは、意外に難しい問題です。

例えば、
 製品Aと、その子 Xがあり、
 部品Xの現在在庫が、0
 リード日数は2日
 基準在庫やロットまとめなどは、考えないとし、
 本日が4月1日だとします。

Aの生産が4月4日と6日に 100 ずつあると、Xの納
 期は、4月2日と4日に 100 100 となります。

	前回		本日					
	1日	2日	3日	4日	5日	6日	7日	8日
A				100		100		100
X		100		100		100		

時が経ち、本日が4月3日になったとします。
 計画通りにXが納品されている場合
 2日にきちんと納入され、本日 所要量計算をすると、
 今 在庫が 100 有り、それを明日 使う計画ですから、
 “計画通り” 特に問題はありません。

Xの納入が遅れている場合
 本日3日に在庫がなくて、明日使うのだから 一見追加
 手配が必要のようですが、ここで 緊急手配するような
 所要量計算をすると、大変なことになってしまいます。
 ちょっと納入遅れがある度に、ダブリ発注です。
 この場合、T P i C S は どう計算するかといいますと、
 “計画に対して、今 遅れが 100 あるのだから、それ
 を本日中に挽回すると 在庫は 100 になり、そうすれ
 ば、明日の生産には問題ない” と考え (?), 追加の手
 配は出ません。

Xが計画以上に納入された場合
 “計画以上の実績”を考えると、T P i C S - III と IV
 では、若干差がありますが、説明を簡単にする為、X
 が、2日に 100 4日に 100 の 注文が既になされ、3
 日の内に、2日の 100 と、4日の 100 が 納入された
 とします。

現在在庫が 200 有る中で所要量計算します。そうすると 6 日の生産分まで在庫があるので 4 日の 100 は、本当はもう要らないはずです。

別の例えで、棚卸の結果余分な在庫が見つかり 在庫数を増やした場合、次の手配は、減らさなければなりません。

しかし、ここで 4 日の計画を変更してしまうことは、次のような意味で出来ません。

① それをすると、後になって計画が全く分からなくなってしまうし、

② この分は 発注済みのものですから、いまさら“いる”とか“要らない”とかと言う問題ではないはずです。

つまり この場合も 計画は、当初のものを変えないようにしなければなりません。

即ち、計画に対し実績が、遅れていても、進み過ぎていても、所要量計算の結果には、影響が表れないように計算しています。

本日終業時の在庫

これを、別の見方で 説明する事ができます。

TP i C S は 所要量計算の時、本日の在庫を基に、明日からの計画を計算します。

所が、本日と言うのが なかなかのくせもので、本日には 24 時間ありますから、工場が始る前と、本日終業後とでは 全く意味が違ってしまいます。

TP i C S は、業務終了後をベースに考えます。

今現在の“現在在庫”を基に、24 時現在の在庫を推定します。

その時 本日までの計画が全て計画通り実行された場合の在庫を計算して出します。

つまり、“現在在庫”に、計画の遅れ進みを逆算して、加味します。

以上の説明は、自分自身の計画に対する遅れ進みだけの話でしたが、生産管理（部品手配）の場合、難しさを倍加しているのが“親との関係”です。

自分が子部品の場合、“現在在庫”には、親の遅れ進みも影響していますから、先程の説明の中に、“現在在庫に 親の遅れ進みも加味し、...”を付け加えなければなりません。

また、自分が製品で 自分の出荷計画から 所要量計算をスタートし、自分の在庫を引当て、自分の生産計画を計算し、子部品に繋げていく場合、出荷実績の遅れ進みも在庫に影響を与えます。

これも同じように、所要量計算の時、加味しなければなりません。

TP i C S では、本日 24 時の在庫を、“理論在庫”と称し、おおよそ 次の計算式で計算します。

$$\begin{aligned} \text{理論在庫} &= \text{現在在庫} + (\text{生産遅れ}) \\ &\quad - (\text{出荷か遅れ}) \\ &\quad - (\text{引き落とし遅れ}) \end{aligned}$$

“遅れ”とありますが、計算は進み過ぎも計算しています。

実績をインプットしない方法

ここで、前回のレポートで最後に触れていた「実績を全くインプットしないで TP i C S を使う方法」にもどります。

先程の例を使い、説明を続けます。

2 日の 100 に対し、100 入庫されてた場合、

$$\text{理論在庫} = 100 + (0) - (0) - (0) = 100$$

です。

では、全く入庫がない場合、

$$\text{理論在庫} = 0 + (100) - (0) - (0) = 100$$

30 だけ 入庫された場合、

$$\text{理論在庫} = 30 + (70) - (0) - (0) = 100$$

現在在庫は 30 になりますが、遅れがまだ 70 残っていますから、理論在庫はやはり 100 になります。

つまり、どのような実績をインプットしても、理論在庫は 常に同じ結果になります。

よって 手配だけを考えるなら、全く実績をインプットしなくても TP i C S を使うことができることになります。

この説明は、なにも数式らしいものを持ってこなくても、先程らしい説明で おおよそ できていた訳ですが、こうして数字を見ると また違った感覚を持つことができるものです。

不良等、在庫の増減の反映

ここまで お読み下さると、こんなことをお考えの方がきつといらっしゃると思います。

「それでは、不良で 在庫を落としたり、棚卸の結果の反映が、できないではないか」

TP i C S は、実績を“計画内の実績”と“計画外の実績”の 2 つに分けて考えていまして、このような動きは、“計画外の実績”にあたります。

先程の、理論在庫の計算 つまり“在庫の補正”は、“計画に対する遅れ進み”を加味していた訳で、計画外の実績については、全く触れていません。

別の言い方をしますと、計画外の実績は、そのままストレートに在庫に反映され、計画内の在庫は、遅れ進みのフィルターを通過してから在庫に反映されます。

ですから、ここでも 先程の例を使って説明しますと、例えば、本日 部品 X を 100 生産したが、その内 30 に

不良が発生し、廃棄処置をした。

この不良の処理についての細かな説明は、TPiCSレポートNo16で行いましたので、ここでは触れませんが、簡単に 部品Xが 一度 100 入庫し、その後“その他出庫”で、30 落としたとします。

現在在庫は、100-30 ですから、70 です。

進捗は、この処理の場合、計画 100 に対し、一応 100 上がっていることにしましたので、遅れ進みは 0 です。

(社外からの調達品の場合、この処理だと 100 分の買掛が発生しますので、この例は 実際には問題があります)

つまり

$$X \text{ の理論在庫} = 70 + (0) - (0) - (0) = 70$$

になります。

この中で所要量計算をしますと、Xは 追加手配するべく計算されます。

このように、計画外の実績と、計画にのっとった実績では、次の所要量計算に全く違った答えが出てきます。

また 同じように、計画外の動きではありますが、

「100 の計画のつもりが、102 出来てしまった」とか、

「98 しか出来なかった」と言うようなケースは、“計画対応実績と在庫対応実績”(別名 たてまえの実績と本音の実績)で処理しますが、この件については次の機会に譲ります。

実績インプットとは何か

遅れ進みの問題はなかなか難しく、導入当初、誰でも一度は悩む“所要量計算の結果が、どうも自分の考えと違う”というケースの、ほとんどの原因が、この問題です。

例えば、No17のレポートでも取り上げた「初期導入時の運営」で、

① 部品、工程の現在在庫をインプットし、② 明日からの製品の計画をインプットし、③ 過去の遅れ進みを全て初期化し、

所要量計算をします。

これからが 今回のテーマに関する話です。

今計算した結果と、TPiCS導入前から生きている実行計画には 必ず差があります。その場合、実行計画で TPiCSの計画を訂正します。

もし、この手続をせず 運営をしていくと、次のような事態に陥ってしまいます。

- ① 初期在庫は正確な数値だったとします。
- ② 従来の実行計画通りに 部品納入 及び生産が出来たとします。
- ③ その実績を正確にインプットしたとします。
- ④ 何日後、現在在庫は正確な数値が維持されています。
- ⑤ 遅れ進みが無く、在庫も正確な数値が維持され、全てが計画通り うまく行っていると思います。

⑥ ここで所要量計算をします。

しかし、TPiCSは 計画に対する遅れ進みが、有ると計算します。

そもそも 対比するべき“計画”が違うのですから当然です。

遅れ進みの認識が違うと言うことは、理論在庫が違うと言うことで、

それは、違う計算結果をもたらすということになります。

かたや、“実績を全くインプットしなくても 正確な所要量計算ができる”と言うのに、

かたや“正確に実績をインプットしても 計算結果がおかしくなる”というのは、面白い結論です。

では、この二つの運営方法の間で、何に違いがあるのかを、もうすこし考えてみます。

その違いは、「計画が生きているか 否か」なのです。

前者の“実績を全くインプットしない方法”の場合でも、立てた計画を無視して生産していたらこの使い方は成り立ちません。

この使い方の場合、“計画を無視する”と言う話は有り得ませんから、必ず 成り立つ訳です。

本来計画を計算させるつもりで、システムを利用するはずですが、皮肉なことに 計画を利用しないで 走ろうとする場合があります。

これが“計画が死んでいる”状態です。

遅れ進みの掴み方

では 次に、この“計画に対する遅れ進み”を、TPiCSは どのように捉えているか?

ご説明します。

この話になると、TPiCS-IIIと TPiCS-IVは、全く違っていて、じつは、IIIとIVの 最大の相違点があるのです。

IIIの場合は、

遅れ進みを“累計値”だけで管理しています。

“本日までの計画累計値”と“本日現在の実績累計値”の差を、“計画との差”とします。

- ① 生産計画に対する遅れ進み、
 - ② 出荷計画に対する遅れ進み、
 - ③ 引き落とし計画に対する遅れ進み
- を、全て累計値の差を持って遅れ進みとしています。

IVの場合は、

注文伝票、作業指示伝票 等、伝票を発行した計画は、部品ごと、日ごとに、発注注番を振り、発注注番ごとに注残データとして持ちます。

実績は、その注番ごとにインプットします。

注残データは、計画日を持っていますので、注残データごとに遅れ進みが分かります。

● 完了	↑ 過去	
● 完了		
○ 未完 (遅れ)		
● 完了		
● 完了		
● 完了 (進み過ぎ)		本日
○ 未完		
● 完了 (進み過ぎ)		
○ 未完		
● 完了 (進み過ぎ)		↓ 未来
○ 未完		
○ 未完		
● 完了 (進み過ぎ)		
○ 未完		
○ 未完		

左図、アミの部分を集めて、“遅れ進み”とします。

このように、ⅢとⅣとは、この実績管理について大きな違いがあります。

当社、TPiCS研修会でも何時も申上げていますが、生産管理 部品手配を 考える場合、“時間”の概念が非常に大事でありまして、時間を“流れ”として常に正確に把握する必要があります。

直線、平面、空間、さらに時間をいれて、常に4次元で考えなくてはなりません。

“時間の流れ”をきちんと掴む為には、その起点である“本日”を、正確に捉え“本日を足掛りに”考えを進めていきます。

今回のテーマのように、在庫や、遅れ進み、というのは、その“考え方”に慣れる為の一番良い題材です。

ハードウェアの障害について

ユーザー数が増えてくると意外に大変なのがハードウェアのトラブルや、環境設定に関するサポートです。TPiCSは ディスク (データ) を頻繁にアクセスするタイプのシステムですから、ハードディスクの故障には とても敏感です。今回のTPiCS-IVのマニュアルでは、ハードウェア 特に、ハードディスクなどのトラブルと思える事態にぶつかった時の、チェックのしかたについて書き加えました。結構 よくある話なので、ここでも、簡単に取上げてみたいと思います。

いやらしいことに、ハードディスクに不具合が発生していても、動作上のエラーメッセージが出ないで、いきなりおかしい計算結果が出されることすらあります。このような事はめったに有りませんが、“最後の可能性”としては、頭の中に置いておく必要があります。

① プログラム域のテスト

ハードディスクに入っているプログラムが、正常であるかどうかをテストします。

リジルのシステムディスクのプログラムを使用して同じ処理をしてみます。

例えば、

使用するシステムがTPiCS-IVで、フロッピーディスクが B : ドライブ、データが A : ドライブのYDT4に入っている場合、
B : ドライブにシステムディスクを入れ、
B > T 4 D = A : YDT 4 ENTER

とします。これにより、使用するプログラムだけを変えて状況を見ることが出来ます。

このテストで正しい結果が出れば、ハードディスク

のプログラム域に不具合があったこととなります。逆に、これでも おかしな結果がでるなら、他の原因による不具合ですから、次のテストに移ります。

② データ域のテスト

データを別の領域にコピーし、そのデータを使って処理をしてみます。

予備のハードディスクがない場合は、臨時的ディレクトリを作り、その中にデータをコピーし、そのディレクトリで処理をしてみます。

ただし、不具合の原因が データの中の論理的エラーである場合は、COPY コマンドでは 論理的エラーもコピーしてしまいますので、テストになりません。また 臨時的ディレクトリを作って テストする場合、従来のディレクトリのデータを抹消しないで下さい。

臨時的ディレクトリ内のデータを作る時、MS-DOS が 従来と同じ場所にデータファイルを作る可能性があるからです。

③ ベリファイによるテスト

VERIFY コマンドは、ディスクにデータを書き込む場合、正常に書き込まれたか否かを チェックしながら処理を続けさせる為のコマンドです。

チェックさせながら TPiCS の処理をします。

④ MS-DOS のファイル管理をする部分のテスト

ここが いかれていると、上記のどのテストをしても結果は皆同じです。

フロッピーディスクの MS-DOS で コンピュータ自身を立上げ、その環境のなかで処理をします。

⑤ SYS. CNT のテスト

SYS. CNT を 一度抹消して、デフォルトの SYS. CNT を 再作成し、処理をします。

⑥ LANのネットワークOSのテスト

スタンドアロンで 使用して見ます。

⑦ その他

解決する場合があります。

MS-DOS の CHKDSK や HDUTY で テストをしてみると
か、極端な話、電源を切って再立上げをするだけで

よく御存知の方から、「二ノ宮さん、何時プログラムを書くのですか？」とか、
逆に、「よくあのレポートまで書く時間がありますね。だって プログラムを作ったり、研修会をやったり、電
話でサポートをしたり、こうしてユーザーと話をしたり...」
などと、言っていただきます。

「いや、それだから出来るのかも分りませんよ。毎日ユーザーと話していると、やはり私にとっても勉強になり
ますし、ましてや 研修会というのは、やる方にとって とても大変なのです。不特定多数の方と 逃げ場の無い
所で 3日間真剣に向い合うのですから。レポートの“ネタ”にはまったく事欠かないんです。

それに、最近 プログラムを書くのが昔と比べますます早くなって来まして、“ワープロでマニュアルを作るよ
り、プログラムを作る方が早い”なんて冗談を言うほどなんです」

ユーザーさんをお願いして、2回だけ「ユーザー事例」を載せさせて頂きましたが、どうもユーザーに負担を掛
けるのが 申し分けなく、連載できないでいます。

先日あるユーザーさんに 「事例」の原稿をお願いしたら、半分真顔でこんな事を言われました。

「TP i CSを使うまでに、当社は これまで いったいどれほどの授業料を払ったことか、それを考えると そう
簡単に他人に教えられませんよ！ やはり数千万円 痛い目に合わないかね、本当の事は分かりませんよ」

そうまで言って頂いて、有難いような 困ったような、複雑な気持です。

これほど激しい会話は 私も初めてですが、TP i CSのユーザーさんは、少なくとも半分以上が、こちらと似たよ
うな経緯ですから、気持は良く分かります。

ご来社下さる方には、しょっちゅう“大変ですよ、大変ですよ”と言っていますが、この問題を避けて通れるなら
その道を選んだ方が良いのではないかと思います。

二ノ宮