



スプレッドシート(注1)統制で重要なことは、リスク・アプローチ

(注2)による評価対象となるスプレッドシートの選定と自動化されたスプレッドシートの判別である。リスク・アプローチは内部統制のすべての実務に共通して、根幹となる考え方である。したがって、スプレッドシート統制も例外ではないという意味で、ここでは殊更これを強調しない。しかし、自動化の判別はスプレッドシート統制独自の要件である。

ところが、この要件を無視すると、たちまちスプレッドシート統制は内部統制の枠組み全体で曖昧な位置付けになってしまう。間違いなく、スプレッドシートはEUC(注3)であるが、電卓の代替のようにマニュアル(手作業)の一部として利用されることが多々ある。これは表計算ソフトが直感的に利用できる簡便性と高度な自動計算を実装できる機能性を同時に有していることに起因する。したがって、スプレッドシート統制はマニュアル統制なのか、あるいは自動化された統制なのか迷うのである。

結論から言えば、個々のスプレッドシートはマニュアル統制で済むものと自動化された統制が必要なものの二

種類に分類できる。しかし、こうした区別なく、スプレッドシート統制だけを独自の枠組みで確立し、個々のスプレッドシートに必要なと思われる統制を逐一リストで一律に実施してしまうような状況を、筆者は「スプレッドシート統制の谷」と呼んでいる。

そもそも、企業が財務報告の信頼

スプレッドシート 統制の谷

マニュアルと自動化の狭間に陥らないために

橋靖宙

新日本監査法人TSR部 シニア

性を確保する上で、何がそれを阻害する要因かを判断せず、「これだけ管理していれば、リスクはないはずだ」というアプローチはSOX法が要請するところではない。また、このアプローチでは、リスクに対してピンポイントで統制を実施する場合に比べ、どうしても広めに強めに統制せざるを得

ず、人手もコストも掛かる。つまり、スプレッドシート統制の谷に陥ってしまうと、真に内部統制を実現できないばかりか、企業経営を圧迫する業務の非効率化も招きかねないのである。

このような状況を回避するためには、先の電卓の代替の例ではマニュアル統制を、またIT固有のリスクが内在するものは自動化された統制を実施するような場合分けが肝要である。このとき、前者と後者を分けるものは第三者による検算可能性という観点である。表計算ソフトの出力(印刷結果やそれと同等の画面表示)上で、スプレッドシートの計算過程を追えるならば、手作業による検算が可能であり、マニュアル統制の対象となる。そうでないならば、何らかの自動計算ロジックが見えない形で実装されており、自動化された統制が必要となる。

さらに、この観点で表計算ソフトの機能に着目すれば、マクロの利用の有無を識別することが実務上でも妥当な境界と言える。マクロはシート上で実現される計算ロジックとは異なり、プログラムとして実現される自動処理だからである。無論、マクロが利用されていても検算可能な場合や、逆

にそうでなくても検算が困難な場合も十分に想定される。実際にマクロの有無だけでなく、ピボットテーブル、他シート参照あるいは条件判定関数の有無を絡めて、検算の困難さを複数の段階で定義する方法もある。つまり、何を検算不可能なリスクと見るかは、利用実態も考慮して、ある程度、主体的な判断に委ねられる。しかし、マクロの自動実行という機性を考えれば、まずマクロの有無を機軸にしてスプレッドシートを振り分けることが、スプレッドシート統制の谷を脱出する第一歩となる。

(注1)このスプレッドシートとは、表計算ソフトが取り扱えるファイル指す。

(注2)環境と目的から判断されるように、対して統制を実施するという考え方。

(注3)End User Computing(エンドユーザーコンピューティング)の略。企業のシステム部門以外の部門が主体的に開発・管理・運用するコンピュータの利用形態。また、そのような形態下のコンピュータシステムやアプリケーション。

