

第18回初級ソフトウェア品質技術者資格試験 問題の解説

※出題した問題の一部を解説付きで公開いたします。正しい知識の習得と理解の深耕にご活用ください。

■問題：

CI（継続的統合）に関する説明として、もっとも適切なものを選び。

【選択肢】

- ア 統合作業は手作業が中心であり、作業手順書による作業ミス防止が重要となる。
- イ 開発工程から本番システムへのデプロイまでを対象としている点が特徴である。
- ウ CIを成功させるためには、ビルドの環境独立性を高めるよう留意する必要がある。
- エ コンパイルやテストを自動化することで、ビルド作業の結果の確認が不要となる。

【正解】

ウ

【出題分野】

この問題は、SQUBOK 樹形図の「第3章 ソフトウェア品質技術」の「3.7 実装の技法」からの出題である。

この問題は、CI（継続的統合）について基本的な考え方を確認する問題である。

【選択肢の解説】

選択肢ア：統合作業を継続的に実施するためには、手作業ではなくツールによる自動化が必要である。したがって「手作業が中心であり」は不適切である。

選択肢イ：CIは、コンパイルやテスト、静的解析といった一連のビルド作業を対象としており、CIの概念をさらに発展させ、開発だけでなく本番システムへのデプロイまで対象を広げた技法としてCD（継続的デリバリー）がある。したがって「デプロイまでを対象としている点が特徴」は不適切である。

選択肢ウ：CIを成功させるためには、特定の統合環境でしかビルドできないような状況を避け、ビルドの環境独立性をできるだけ高める必要がある。したがってこの選択肢は適切である。

選択肢エ：CIツールによるコンパイルやテストの自動化で、結果の確認が不要になることはなく、CIツールが保持している実行結果を確認し品質状況を開発者にフィードバックすべきである。したがって「結果の確認が不要となる」は不適切である。

【解説】

CI（継続的統合）とは、あらかじめ定めた時刻や、バージョン管理ツール上のソフトウェア成果物に変更があるたびに、コンパイルやテスト、静的解析といった一連のビルド作業を実施し、ソフトウェアを常に統合することである。CIを実現することで、障害を早期に検出できることや、ビルド自動化により手作業に起因するミスを低減できること、品質を断片的ではなく時系列に可視化できることなどの効果が期待できる。

CI環境を構築するには、バージョン管理ツール、ビルドツール、テストツール、CIツールの連携が必要である。CIツールがバージョン管理ツールのリポジトリを監視し、変更があるたびにビルドツールを起動し、自動的にコンパイル、テスト等を実行する。実行結果を集計し、開発者にビルドの成否をはじめとする品質状況をレポートする。

CIを成功させるためには、継続して動作し続けられるようにすることが重要で、特定の統合開発環境でしかビルドできないというような状況を避け、環境独立性の高いビルドツール上でビルドできるように整備する必要がある。また、一部のプログラム変更によりテストが失敗しても全体が失敗することがないようにスタブを用意するなど、テスト実行を継続させる工夫を組み込んでおくことなども必要である。

CIの概念を発展させたものとして、CD（継続的デリバリー）があり、本番・運用環境へのデプロイまでを対象としている。また、より広いフェーズを自動化し、開発（Development）と運用（Operations）全体のサイクルを短時間で回すことで、ビジネス要求に対してより柔軟にスピーディに対応できるシステムを作り上げるための技法としてDevOpsが注目されている。

■問題

GQM手法に関する説明として、もっとも適切なものを選べ。

【選択肢】

- ア GQM手法は、測定可能なメトリクスから目標へと対応付けるボトムアップな技法である。
- イ GQM手法の適用結果は、目標－品質－メトリックの階層的な対応付けで構成される。
- ウ GQM手法の適用により、一つの目標に対応した唯一のメトリックが導かれる。
- エ GQM手法の適用により、収集すべきデータが明確になる。

【正解】

エ

【出題分野】

この問題は、SQuBOK 樹形図の「第3章 ソフトウェア品質技術」の「3.1 メトリクス」からの出題である。この問題は、測定項目であるメトリクスを評価目標に対応させるための枠組みであるGQM手法の基本的な考え方を確認する問題である。

【選択肢の解説】

測定はトップダウンで行われることが望ましい。まず測定の目標があり、その目標を遂行するためにメトリクスが定義され、実際に測定されることが望ましい。

選択肢ア：GQM手法は、目標から測定項目であるメトリクスをトップダウン的に決定するための手法である。

したがって、測定可能なメトリクスから目標へと対応付ける「ボトムアップな技法」は不適切である。

選択肢イ：GQM手法は目標（Goal）層、質問（Question）層、メトリック（Metric）層の3層構造で構成されるモデル化手法である。したがって、GQM手法の適用結果は、目標－質問－メトリックの階層的な対応付けで構成される。アルファベットのQは品質（Quality）の意味で使われることが多いが、GQM手法のQは質問（Question）の意味で使われている。したがって、「目標－品質－メトリックの階層的な対応付け」は不適切である。

選択肢ウ：GQM手法の3層の各要素間の対応関係は多対多の関係にある。ひとつの目標は複数の質問に対応づけることもできるし、各質問は複数のメトリクスに基づいて決定されることもできる。したがって、「一つの目標に対応した唯一のメトリックが導かれる」という記述は不適切である。

選択肢エ：GQM手法を利用すると、測定目標とメトリクスの対応関係が明確になり、データを収集する目的の確認が容易となり、収集すべきデータが明確になる。よって、この選択肢は適切である。

【解説】

GQM (Goal-Question-Metric) 手法は、メリーランド大学のビクター・バシリ教授 (Victor R. Basili) らによって開発された、測定の手組み及びモデル化手法である。この手法を利用すると、測定目的・目標とメトリクスの対応関係が明確になり、データを収集する目的・目標の確認が容易となる。

ソフトウェア開発に関連するあらゆる測定は、目的なしにメトリクスやデータを集めるのではなく、何のために測定するのか、前提となるモデルは何か、結果をどう解釈するかなどを明確にした上で実施されるべきである。GQM手法はこれらを定義する代表的な手組みのひとつであり、測定を以下の3つの層で定義する。

(1) 目標層 (Goal 層)

測定を行う目標を定義する。測定対象、測定理由、測定観点・視点、測定環境等に基づいて明確にする。

(2) 質問層 (Question 層)

目標の達成のために評価すべき質問を定義する。特定の目標が達成されたか否かを判断する具体的な基準となる。ここでは、目標を評価または達成するための方法を質問の集合を用いて特徴づける。

(3) メトリック層 (Metric 層)

前提となるモデルに基づいて測定可能なメトリクスを定義する。上記の質問に定量的な回答を提供するために必要な情報の集合である。この情報は客観的データと主観的データに分類できる。作業工数やプログラムサイズなどのように形式的に定義でき測定方法もある程度確立されている対象に対しては、客観的データを収集する。顧客満足度のレベルなどのように形式的な定義が現段階では難しく、測定方法も確立されていない対象に対しては、主観的データを収集する。主観的データの場合は、測定観点の理解に依存するため、測定者によって差異が出る危険性があることに注意が必要である。

一般に、各層の要素間の対応関係は多対多の関係にある。すなわち、ひとつの目標は複数の質問に結び付けられ、各質問への回答は複数のメトリクスに基づいて決定される。逆に、ひとつのメトリックを複数の質問のために使用することができる。

