



Justification of the Six WQI Core Management System Quality Factors

WHOLE QUALITY JUSTIFICATION WQI_FMJ1

Methodological Justification of the Six-Factor Architecture of the
WQI Management System Quality Standard — Core

Article — September 2026

WHOLE QUALITY INSTITUTE

Table of Contents

FOREWORD.....	4
1. PURPOSE AND SCOPE	4
1.1 Purpose of This Paper.....	4
1.2 What Is Being Justified.....	4
1.3 Scope of the Justification	5
2. MANAGEMENT SYSTEM QUALITY DETERMINATION TASK.....	5
2.1 Management as the Subject of Quality Determination	5
2.2 Management System Intended Function.....	6
2.3 Management System Intended Result	6
2.4 Relationship Between Management System Quality and the Managed Result	7
3. METHODOLOGICAL RESOLUTION.....	7
3.1 Need for Appropriate Core Resolution	7
3.2 Core Factors and Detailed Management Activities	8
3.3 Stability Across Different Management Contexts	8
3.4 Avoiding Premature Detail.....	8
4. METHOD FOR DERIVING CORE QUALITY FACTORS.....	9
4.1 Derivation from Function Realization	9
4.2 Identification of Functional Failure-Mode Families	10
4.3 Test for a Separate Core Factor.....	10
4.4 Necessity and Removal Test.....	10
4.5 Distinctness and Non-Redundancy Test	11
4.6 Cross-Context Applicability Test	11
4.7 Evidence and Determinability Test.....	12
4.8 Collective Architecture Test.....	12
5. WHY THE ARCHITECTURE RESOLVES INTO SIX FACTORS	13
5.1 From Intended Function to Six Functional Areas	13
5.2 Why Requirements Realization Must Remain Separate from Realization Capability	14
5.3 Why Quality-State Determination Must Remain Separate from Realization Capability	14
5.4 Why Quality-State Information Must Remain Separate from Quality-State Determination	14
5.5 Why Quality-State Influence Must Remain Separate from Determination and Information ...	15
5.6 Why Lifecycle and Interface Realization Must Remain Separate.....	15
5.7 Why Six Functional Areas Form One Architecture	16
6. FACTOR-BY-FACTOR JUSTIFICATION.....	16
6.1 Factor 1 - Requirements Realization	16
6.2 Factor 2 - Realization Capability	17

6.3 Factor 3 - Quality-State Determination	18
6.4 Factor 4 - Quality-State Information	18
6.5 Factor 5 - Quality-State Influence.....	19
6.6 Factor 6 - Lifecycle and Interface Realization	20
7. COLLECTIVE TESTING	21
7.1 Completeness Test	21
7.2 Distinctness and Overlap Test	21
7.3 Removal Test	22
7.4 Non-Compensation and Failure-Visibility Test	22
7.5 Consistency of Methodological Resolution.....	23
7.6 Collective Finding	23
8. CROSS-DOMAIN STRESS TESTING.....	24
8.1 Purpose of Cross-Domain Stress Testing	24
8.2 Different Types of Managed Results	24
8.3 Different Realization Methods and Technologies	25
8.4 Different Organizational and Provider Arrangements.....	25
8.5 Different Lifecycle Conditions and Change	26
8.6 Normal, Unusual, and Degraded Conditions.....	26
8.7 Cross-Domain Stress-Test Finding	27
9. WHY SIX, AND NOT FEWER OR MORE.....	27
9.1 Why Fewer Factors Would Reduce Determination Capability	27
9.2 Why Additional Factors Are Not Required at the Present Core Methodological Resolution ..	28
9.3 Why Six Is the Supported Core Resolution	28
10. STABILITY AND CONTROLLED REVISION.....	29
10.1 Stability of the Core Factor Architecture.....	29
10.2 Conditions That Could Justify Revision	30
10.3 Controlled Revision Principle	31
10.4 Status of the Present Architecture.....	32
11. CONCLUSION	33
Copyright and Use Notice	35

FOREWORD

Methodological justification of the six-Factor architecture established for Management System Quality Determination under WQI_QMS1.

Management is used in many organizations and activities to turn requirements into products, services, work, decisions, actions, and other Intended Results.

A Management System may include policies, procedures, responsibilities, people, information, resources, technologies, and controls. The existence of these elements alone, however, does not establish how well management actually works.

WQI_QMS1 examines the quality of the actual Management System through six Core Quality Factors: Requirements Realization; Realization Capability; Quality-State Determination; Quality-State Information; Quality-State Influence; and Lifecycle and Interface Realization.

This creates a fundamental methodological question: Why these six Factors?

A Core architecture should not be accepted merely because its components have been named. Each Factor requires a reason for being separately visible, and the Factors collectively require a reason for being treated as a sufficiently complete architecture at the selected Methodological Resolution.

This paper therefore examines how Management System Function Realization can succeed or fail; whether each proposed Factor represents a necessary and distinguishable functional area; what determination capability would be lost if a Factor were removed; whether Factors can be combined without loss of important information; and whether additional Core Factors are required.

The purpose is not to prove that six Factors are correct because WQI_QMS1 already uses six Factors. The purpose is to determine whether the six-Factor architecture can be methodologically justified.

The analysis proceeds step by step from the Management System Intended Function and Intended Result to the presently supported Core Factor architecture.

1. PURPOSE AND SCOPE

1.1 Purpose of This Paper

WQI_QMS1 uses six Core Quality Factors to examine Management System Quality.

The purpose of this paper is to explain why these six Factors are needed and why they provide an appropriate Core structure at the present Methodological Resolution.

The starting point is not the number six. The starting point is the practical determination question: What must be examined to understand how well management actually works?

A Core architecture must provide enough detail to distinguish important management functions and make significant weaknesses visible. At the same time, it should not become a detailed checklist of individual practices, controls, or activities.

This paper therefore examines whether each Factor is necessary, what would be lost if it were removed, whether Factors can be combined without loss of important determination capability, and whether additional Factors are needed.

The purpose is not to defend a predetermined architecture. It is to determine whether the six-Factor structure established in WQI_QMS1 can be methodologically justified.

1.2 What Is Being Justified

The architecture being justified consists of six Management System Quality Factors:

Requirements Realization

Realization Capability

Quality-State Determination

Quality-State Information

Quality-State Influence

Lifecycle and Interface Realization

These Factors identify major parts of Management System Quality that need to remain visible during Quality Determination.

A Factor is not simply an important management subject. Policies, procedures, resources, competence, communication, planning, monitoring, change, and many other subjects can be important without each becoming a separate Core Factor.

The fundamental question is: Does the proposed Factor represent a distinct part of how management achieves, or can fail to achieve, what is required?

A Core Factor should therefore be retained only if its removal would leave an important part of Management System Function Realization insufficiently visible and that loss could not be adequately recovered through the remaining Factors at the same Core Methodological Resolution.

The Factors must also work together, preserving important differences without unnecessary division or duplication.

This paper therefore justifies both the individual necessity of the Factors and their adequacy as a connected Core architecture.

It does not require organizations to use the same management arrangements, methods, technologies, or detailed requirements.

1.3 Scope of the Justification

This paper provides a methodological justification of the six Core Management System Quality Factors.

The analysis examines whether the Factors arise from the Management System Intended Function; represent distinguishable parts of Function Realization; make significant strengths, weaknesses, and failures visible; remain sufficiently distinct from one another; collectively provide sufficient Core coverage; and remain meaningful across materially different management contexts.

The justification concerns the architecture used to determine Management System Quality. It does not prescribe how a Management System should be designed or operated.

The paper does not establish organization-specific procedures, targets, assessment methods, metrics, or Quality Outcome Criteria.

It also does not determine the Quality State of any particular Management System.

The detailed justification of the individual Core Quality Indicators is addressed separately in the WQI Management System Quality Indicators Justification.

The central question of this paper is therefore:

Why does Management System Quality Determination require these six Core Quality Factors?

2. MANAGEMENT SYSTEM QUALITY DETERMINATION TASK

2.1 Management as the Subject of Quality Determination

The subject of this determination is the actual management used by an organization to realize a required product, service, work, or other Intended Result.

The Management System and the managed result are distinct Quality Objects.

A required result may sometimes be achieved despite weaknesses in management. Conversely, management may adequately perform some of its functions while the resulting condition is affected by circumstances outside its applicable responsibility or authority.

Management System Quality therefore cannot be determined solely from the final managed result.

The determination must examine how the Management System performs the functions necessary to ensure, within its applicable responsibility and authority, that the required result is realized as required.

The Core Quality Factors provide the major distinguishable parts of this management function that must remain visible during that determination.

2.2 Management System Intended Function

The Management System Intended Function is:

“To ensure, within the applicable responsibility and authority of the Management System, that the required product, service, work, or other Intended Result is realized as required.”

This definition establishes an important boundary.

To ensure does not mean that the Management System guarantees every outcome or exercises unlimited control over conditions outside its responsibility and authority.

The phrase “realized as required” connects the Management System with requirements, realization activities, capability, information, determinations, management actions, and the relationships through which realization occurs.

Each Core Quality Factor must therefore connect to a necessary and distinguishable part of this Intended Function.

If a proposed Factor cannot be connected to Management System Function Realization, its place in the Core architecture should be reconsidered.

Conversely, if an essential part of Function Realization cannot be adequately examined through the Factor architecture, the architecture may be incomplete.

The Management System Intended Function therefore provides the first methodological basis for deriving and testing the Core Factors.

2.3 Management System Intended Result

The Management System Intended Result is:

“The required product, service, work, or other Intended Result is realized as required within the applicable responsibility, authority, Boundaries, Interfaces, Context, and temporal basis of the Management System.”

This definition further establishes the conditions within which Management System Quality is determined.

Responsibility and authority define the management scope within which the Management System can be expected to act.

Boundaries and Interfaces identify where management functions begin, end, connect, transfer, or depend on other functions or organizations.

Context identifies the conditions under which Function Realization occurs.

The temporal basis recognizes that Management System Quality is not independent of time. Requirements, capability, information, actual conditions, responsibilities, and Interfaces can change.

The same management arrangement may therefore perform differently as these conditions change. The Core Factors must be capable of examining not only individual management activities but also their connected realization under these applicable conditions.

2.4 Relationship Between Management System Quality and the Managed Result

Management System Quality and the quality of the managed result are related, but they are not the same determination.

The managed product, service, work, or other Intended Result is one Quality Object.

The Management System used to realize that result is another Quality Object.

Each has its own Intended Function, Intended Result, Boundaries, Context, applicable Quality Outcome Criteria, and Evidence.

A satisfactory managed result does not by itself establish satisfactory Management System Quality.

An unsatisfactory managed result does not by itself establish poor Management System Quality, particularly where relevant conditions are outside the Management System's applicable responsibility or authority.

The connection between the two Quality Objects must nevertheless be preserved.

Management System Quality concerns whether the actual Management System performs its Intended Function in relation to what must be realized.

The quality of the managed result concerns whether that result realizes its own Intended Function and Intended Result.

The Management System Quality Factors must therefore allow management to be examined as a distinct Quality Object without substituting the Quality State of the managed result for the Quality State of the Management System.

3. METHODOLOGICAL RESOLUTION

3.1 Need for Appropriate Core Resolution

A Management System can be described at many levels of detail.

At a very broad level, management might be judged simply as effective or ineffective. Such a resolution is too coarse for the WQI Core because important weaknesses can remain hidden inside a single overall judgment.

At the opposite extreme, every management activity, procedure, role, record, control, and decision could be treated as a separate determination subject. Such a resolution would be too detailed for a stable cross-context Core architecture.

The Core therefore requires an appropriate Methodological Resolution.

At this resolution, management functions should be separated where their differences are necessary for meaningful Quality Determination and grouped where further separation would primarily describe detailed implementation.

The objective is not to list everything management does.

The objective is to identify the major distinguishable parts of Management System Function Realization that must remain visible.

A Core Factor should therefore be broad enough to remain meaningful across different contexts but specific enough that its removal would conceal an important part of management performance.

The six-Factor architecture can be justified only if it provides this balance.

3.2 Core Factors and Detailed Management Activities

Core Quality Factors are not equivalent to individual management activities, practices, or controls.

Planning, training, provider selection, equipment maintenance, reviews, inspections, change activities, communications, and many other management activities may contribute to Management System Function Realization.

These activities can provide Evidence about Management System Quality without each becoming a separate Core Factor.

A Core Factor groups related management functions at a level that remains useful across different applications.

The specific activities used to realize those functions may vary substantially, allowing the Core architecture to remain stable while application adapts to different organizations, technologies, managed results, and conditions.

The Factor architecture therefore does not prescribe one preferred form of management.

It identifies the major parts of management performance that need to remain visible during Quality Determination.

3.3 Stability Across Different Management Contexts

A Core Factor should remain meaningful when the organization, managed result, technology, operating environment, or realization method changes.

The applicable requirements, risks, procedures, technologies, Evidence, and Quality Outcome Criteria may differ substantially among contexts.

The Core functional structure should remain sufficiently stable beneath these differences.

A subject that is important only because of one domain-specific practice may belong at a context-specific level rather than as an invariant Core Factor.

A functional area that repeatedly remains necessary as contexts change provides stronger support for inclusion in the Core architecture.

Cross-context stability is therefore an important test of the proposed Factors.

This does not mean that every Factor must be applicable in exactly the same way in every determination.

Where a corresponding management function or dependency does not exist, applicability may differ.

The requirement is that the Factor remain meaningful wherever the corresponding function is necessary.

3.4 Avoiding Premature Detail

The Core architecture should identify major management functions before detailed determination questions are introduced.

If the architecture begins with specific practices, procedures, technologies, or records, it can become dependent on the context from which those details were taken.

The derivation therefore begins with a broader question:

What distinct parts of management must remain visible to determine whether the Management System realizes its Intended Function?

Only after those parts are established should more detailed Core Quality Indicators be used to identify what should be examined within each Factor.

The resulting methodological hierarchy is:

Management System Intended Function and Intended Result

↓

Management System Function Realization

↓

Core Quality Factors

↓

Core Quality Indicators

↓

Applicable Quality Outcome Criteria

↓

Evidence

↓

Evidence-Supported Findings

The Factors therefore occupy an intermediate resolution.

They are more specific than an overall judgment of Management System Quality but broader than the individual determination functions represented by the Core Quality Indicators.

This position helps prevent the Core architecture from becoming either too broad to reveal meaningful weaknesses or too detailed to remain stable across contexts.

4. METHOD FOR DERIVING CORE QUALITY FACTORS

4.1 Derivation from Function Realization

The Core Factors are derived from Management System Function Realization rather than from common management categories, organizational departments, or standard lists of procedures.

The derivation begins by asking what management must be able to do if its Intended Function is to be realized.

Management must know what is required.

Management must possess or establish the capability needed to realize what is required.

Management must be able to determine the actual condition relevant to its responsibilities.

Quality-relevant information must remain usable by the management functions that depend on it.

Management must be able to act appropriately when a condition requires management influence.

These functions must remain connected across relevant Lifecycle stages, changes, transfers, handoffs, and Interfaces.

Identifying these functional needs does not by itself justify them as separate Core Factors.

Each must be tested for necessity, distinctness, removal consequence, determinability, and cross-context stability.

The Factor architecture is therefore derived from the Quality Determination task rather than from a predetermined list of management subjects.

4.2 Identification of Functional Failure-Mode Families

A useful way to test the emerging architecture is to examine how Management System Function Realization can fail.

Management can fail because requirements are incomplete, unclear, incorrectly connected, improperly allocated, inadequately verified, incorrectly changed, or not maintained.

It can fail because the capability needed for realization is absent, inadequate, unavailable, or lost after change or under unusual conditions.

It can fail because the actual condition is not adequately determined.

It can fail because Quality-relevant information is unavailable, disconnected, outdated, incorrectly communicated, insufficiently protected, or otherwise unusable.

It can fail because management cannot appropriately protect, contain, stabilize, respond, intervene, or address an affected condition.

It can also fail because necessary functions are lost or disconnected across Lifecycle stages, changes, transfers, handoffs, or Interfaces.

These represent distinguishable families of functional failure.

A Management System may be strong in one family while weak in another.

For example, strong realization capability does not correct an incomplete requirement.

Accurate Quality-State information does not establish intervention authority.

Effective local management action does not by itself ensure that requirements or Quality-State information remain connected across an organizational Interface.

The presence of distinguishable failure-mode families therefore supports separate Core Factors where combining them would hide materially different weaknesses.

4.3 Test for a Separate Core Factor

A proposed Core Factor should satisfy several methodological tests.

First, is the represented function necessary for Management System Function Realization?

Second, does it create a distinguishable Quality Determination question?

Third, would combining it with another Factor hide a materially different management weakness?

Fourth, can the represented condition be meaningfully examined through Evidence?

Fifth, does the function remain relevant across materially different management contexts?

Sixth, does the Factor add necessary determination information without unnecessary duplication?

A separate Core Factor is supported when the represented function is necessary, distinguishable, Evidence-determinable, sufficiently stable across contexts, and non-redundant within the collective architecture. Distinctness does not require isolation: a Factor may depend on or contribute to other Factors while retaining its own determination purpose.

4.4 Necessity and Removal Test

The necessity of a proposed Factor can be examined by removing it from the architecture and asking what determination capability would be lost.

The central removal question is:

“If this Factor were removed, could an important weakness in Management System Function Realization remain hidden while the remaining Factors appeared satisfactory?”

If Requirements Realization were removed, management capability could appear satisfactory even though management was realizing incomplete or incorrect requirements.

If Realization Capability were removed, requirements could be well established while the actual means to realize them remained inadequate.

If Quality-State Determination were removed, management could have requirements and capability without adequately knowing the actual condition.

If Quality-State Information were removed, a condition could be correctly determined but the information could fail to remain usable by the functions that depend on it.

If Quality-State Influence were removed, management could know about an adverse condition without having the ability to appropriately influence it.

If Lifecycle and Interface Realization were removed, local functions could appear satisfactory while necessary connections failed between stages, organizations, or responsibilities.

A Factor is therefore necessary when its removal creates a meaningful blind area that cannot be adequately covered by the remaining Factors without changing their defined purposes or moving them to a different Methodological Resolution.

4.5 Distinctness and Non-Redundancy Test

The Factors are expected to interact, but interaction does not make two Factors redundant.

The distinctness test asks whether satisfactory realization of one Factor necessarily establishes satisfactory realization of another.

If it does not, the Factors may need to remain separate.

Requirements can be adequately established while realization capability is inadequate.

Realization capability can be strong while the actual condition is not adequately determined.

A condition can be correctly determined while the resulting information is unavailable or disconnected.

Information can be available while management lacks the ability or authority to appropriately influence the condition.

Local management functions can perform adequately while their connections across Lifecycle stages or Interfaces fail.

Non-redundancy therefore does not require complete independence.

It requires that each Factor preserve a determination purpose that is not fully represented by another Factor.

4.6 Cross-Context Applicability Test

A Core Factor should remain meaningful across materially different management contexts.

The management of manufacturing, services, engineering, construction, maintenance, technology, professional work, or multi-organization realization may use very different procedures, technologies, Evidence, and Quality Outcome Criteria.

The functional question represented by a Core Factor should nevertheless remain recognizable.

The cross-context test therefore asks whether the represented management function remains meaningful as the realization context changes.

A subject whose importance depends primarily on one particular management practice may be too specific for Core Factor status.

This does not require universal applicability in every individual determination.

Where the corresponding function or dependency does not exist, a particular application may establish that the Factor or a lower-level Indicator is not applicable.

Applicability must be established rather than assumed.

The purpose of the test is to distinguish a stable Core functional area from a context-dependent management practice.

4.7 Evidence and Determinability Test

A Core Factor must be capable of meaningful examination through Evidence.

It is not sufficient for a proposed Factor to represent an important idea.

The determination methodology must be able to ask:

“Can the condition represented by the proposed Factor be examined through relevant Evidence and support Evidence-Supported Findings about Management System Function Realization?”

Requirements Realization may be examined through requirements, allocations, changes, verification records, and related Evidence.

Realization Capability may be examined through Evidence concerning methods, processes, providers, personnel, equipment, infrastructure, and applicable conditions.

Quality-State Determination may be examined through determination scope, methods, competence, Evidence use, criteria, interpretation, and current validity.

Quality-State Information may be examined through information identity, connections, status, communication, availability, and protection.

Quality-State Influence may be examined through Evidence of protection, containment, response, authority, intervention, resulting condition, affected scope, and cause or recurrence where required.

Lifecycle and Interface Realization may be examined through responsibilities, transfers, handoffs, requirement connections, Quality-State continuity, change effects, and feedback.

The specific Evidence will depend on Context.

Examination under the Factor must nevertheless support bounded Evidence-Supported Findings at the applicable determination resolution.

Evidence Quality remains a separate determination concern.

The existence of Evidence does not establish that the Evidence is of sufficient quality for the Finding it is intended to support.

A proposed Factor that cannot be meaningfully determined through Evidence would be insufficiently defined for inclusion in the Core architecture.

4.8 Collective Architecture Test

Individual Factors can be useful while the collective architecture remains incomplete, repetitive, or inconsistent.

The proposed Factors must therefore also be tested together.

The collective architecture test asks whether the Factors provide sufficient coverage of Management System Function Realization; preserve important failure visibility; maintain meaningful distinctions; avoid unnecessary duplication; operate at a consistent Methodological Resolution; remain usable across different contexts; and avoid treating one Factor as an automatic substitute for another.

The Factors are connected, but they are not automatically compensatory.

Strong Realization Capability does not compensate for inadequate Requirements Realization.

Accurate Quality-State Determination does not compensate for unusable Quality-State Information.
Good information does not compensate for inability to appropriately influence an adverse condition.
Adequate local functions do not compensate for loss of necessary continuity across an Interface.

The collective test also asks whether an additional Core Factor is needed or whether two proposed Factors can be combined without meaningful loss.

The architecture is supported only if the Factors are justified individually and collectively.

The next section applies this reasoning to explain why the architecture resolves into six Core functional areas.

The tests are intentionally cumulative. Necessity alone is insufficient for separate Factor status; a function must also remain distinct, non-redundant, determinable through Evidence, cross-context meaningful, and coherent with the collective architecture at the same Core Methodological Resolution.

5. WHY THE ARCHITECTURE RESOLVES INTO SIX FACTORS

5.1 From Intended Function to Six Functional Areas

The derivation begins with the Management System Intended Function:

“To ensure, within the applicable responsibility and authority of the Management System, that the required product, service, work, or other Intended Result is realized as required.”

For this function to be meaningfully examined, six major management needs must remain visible.

Management must establish and maintain what is required. This resolves into Requirements Realization.

Management must establish and maintain the capability needed to realize what is required. This resolves into Realization Capability.

Management must be able to determine the actual condition relevant to its responsibilities. This resolves into Quality-State Determination.

Quality-relevant information must remain sufficiently usable and connected for Management System Function Realization. This resolves into Quality-State Information.

Management must be able to appropriately influence a condition when management action is required. This resolves into Quality-State Influence.

Management System Function Realization must remain sufficiently connected through relevant Lifecycle stages, changes, transfers, handoffs, and Interfaces. This resolves into Lifecycle and Interface Realization.

The resulting Core Factors are:

Requirements Realization

Realization Capability

Quality-State Determination

Quality-State Information

Quality-State Influence

Lifecycle and Interface Realization

The number six is not the design objective.

It results from separating the major functional needs that must remain visible at the selected Core Methodological Resolution.

The following sections explain why these functional areas should remain separate.

5.2 Why Requirements Realization Must Remain Separate from Realization Capability

Requirements Realization and Realization Capability answer different determination questions.

Requirements Realization concerns what must be realized.

Realization Capability concerns whether management has established and maintained the capability needed to realize it.

A Management System can be highly capable in technical or operational terms while acting against incomplete, incorrect, outdated, or improperly allocated requirements.

Conversely, requirements can be complete and correctly established while the methods, processes, providers, personnel, equipment, infrastructure, or conditions needed for realization are inadequate.

Combining these functions would therefore hide two materially different failure modes.

Removing Requirements Realization would allow capability to appear satisfactory even when management is realizing the wrong or incomplete requirement.

Removing Realization Capability would allow well-established requirements to appear sufficient even when management lacks the practical ability to realize them.

The two Factors interact closely, but neither substitutes for the other.

They therefore remain separate at the Core Methodological Resolution.

5.3 Why Quality-State Determination Must Remain Separate from Realization Capability

Realization Capability concerns the ability to realize what is required.

Quality-State Determination concerns the ability to determine the actual condition relevant to management responsibilities.

The existence of adequate realization capability does not establish that management knows the actual condition.

A process may be capable, personnel may be competent, and equipment may be suitable while the actual condition remains inadequately observed, measured, assessed, interpreted, or concluded.

Quality-State Determination requires its own consideration of scope, method, coverage, assessment capability, competence, Evidence use, applicable Quality Outcome Criteria, interpretation, Quality-State conclusion, and current validity.

The reverse distinction also matters.

A Management System may be able to determine an adverse condition accurately while lacking the capability needed to realize the required result.

Combining the two would therefore obscure the difference between being able to realize a requirement and being able to determine the actual condition.

Quality-State Determination must remain separately visible.

5.4 Why Quality-State Information Must Remain Separate from Quality-State Determination

Quality-State Determination concerns establishing an Evidence-supported understanding of the actual condition.

Quality-State Information concerns whether Quality-relevant information remains sufficiently identifiable, connected, current, available, communicated, and protected for Management System Function Realization.

A determination can be correct while the resulting information fails to reach a responsible function.

A record can exist while being disconnected from the relevant requirement, subject, Lifecycle stage, or change history.

Information can become outdated, unavailable, incorrectly communicated, or insufficiently protected.

These failures can occur even when the original Quality-State Determination was adequate.

The reverse is also possible.

A well-designed information system does not correct an inadequate determination of the actual condition.

The two Factors therefore represent different management functions.

Quality-State Determination establishes what management can determine about the condition.

Quality-State Information examines whether the information needed for Function Realization remains usable and connected.

They must remain separate.

5.5 Why Quality-State Influence Must Remain Separate from Determination and Information

Knowing an actual condition and communicating it are not sufficient when management action is required.

A Management System may correctly determine an adverse condition and make the information available to responsible functions while still lacking the authority, capability, timing, or response needed to appropriately influence that condition.

Quality-State Influence therefore addresses a different management function.

It concerns the degree to which management can appropriately protect, contain, stabilize, respond, intervene, establish an acceptable resulting condition, address affected scope, and address cause and recurrence where required.

The distinction can be stated simply:

Knowing that a problem exists does not correct it.

Information about a required action does not establish the ability or authority to perform that action.

Quality-State Influence must therefore remain separate from Quality-State Determination and Quality-State Information while remaining functionally connected to both.

5.6 Why Lifecycle and Interface Realization Must Remain Separate

Management System Function Realization often extends across multiple stages, functions, responsibilities, organizations, providers, changes, and transfers.

Individual management functions can perform adequately within their local scope while the overall realization fails at the connections between them.

A requirement can be correctly established but incorrectly transferred.

A Quality State can be adequately determined but not preserved or confirmed at a later Lifecycle stage.

Responsibility can be clear within two organizations while becoming unclear at their Interface.

A change can be correctly managed locally while its downstream effects are not adequately connected.

Lifecycle and Interface Realization therefore asks a distinct question:

Do the necessary management functions remain sufficiently connected as realization passes through relevant stages, changes, transfers, handoffs, and Interfaces?

This function is not fully represented by Requirements Realization, Realization Capability, Quality-State Determination, Quality-State Information, or Quality-State Influence individually.

It addresses a distinct class of weakness: loss of functional continuity between otherwise adequate parts.

It must therefore remain separately visible.

5.7 Why Six Functional Areas Form One Architecture

The six Factors remain distinct for determination purposes while forming one connected architecture.

Their relationship can be expressed through a simplified explanatory sequence:

What is required?

↓

Can it be realized?

↓

What is the actual condition?

↓

Is Quality-relevant information usable?

↓

Can management appropriately influence the condition where needed?

↓

Do these functions remain connected across relevant Lifecycle stages and Interfaces?

This sequence is explanatory rather than strictly chronological.

In actual Management System Function Realization, the functions can operate repeatedly, concurrently, and through feedback loops.

Requirements can change after new information is obtained.

Capability can be modified after a Quality-State Determination.

Management influence can create a new condition requiring further determination.

Lifecycle and Interface relationships can affect every other Factor.

The architecture therefore represents interacting but non-substitutable functional areas.

Together they provide the basis for examining Management System Function Realization without reducing management to either a single overall judgment or a detailed list of individual practices.

The next section examines each Factor individually against the derivation tests.

6. FACTOR-BY-FACTOR JUSTIFICATION

6.1 Factor 1 - Requirements Realization

The Core purpose of Requirements Realization is:

“The degree to which management establishes, connects, allocates, verifies, changes, and maintains the requirements necessary to define what must be realized.”

Management cannot reliably realize what is required if what must be realized has not been adequately established and maintained.

Requirements provide the substantive basis for realization.

Requirements Realization is not limited to the existence of requirement documents.

A requirement may exist but be incomplete, unclear, disconnected from the Intended Result, incorrectly allocated, inadequately verified, incorrectly changed, or lost when continuing applicability is required.

The Factor therefore asks a distinct management question:

Has management adequately established and maintained what must be realized?

If this Factor were removed, a Management System could appear capable and well controlled while operating against incomplete or incorrect requirements.

The resulting weakness would not be adequately represented by Realization Capability or the later Quality-State functions.

Requirements Realization is also distinct from the Quality State of the managed result.

The Factor examines how management handles the requirements necessary for Function Realization, not whether the managed result itself ultimately satisfies those requirements.

The detailed Core Quality Indicators beneath this Factor provide the next level of determination resolution.

At the Factor level, the methodological finding is that Requirements Realization is necessary, distinguishable, Evidence-determinable, and meaningful across different management contexts.

It is therefore justified as a separate Core Quality Factor.

6.2 Factor 2 - Realization Capability

The Core purpose of Realization Capability is:

“The degree to which management establishes and maintains the capability needed to realize what is required under applicable conditions, including after change and under unusual conditions.”

Knowing what is required does not establish the practical ability to realize it.

Management must establish and maintain suitable realization methods, processes, providers, personnel competence, equipment, supporting infrastructure, and applicable realization conditions.

Capability must also remain adequate when conditions change.

A capability that was adequate before a change may no longer be adequate afterward.

Similarly, a Management System may perform adequately under normal conditions while lacking the capability required under applicable unusual conditions.

Realization Capability therefore asks:

Does management establish and maintain the capability needed to realize what is required?

If this Factor were removed, requirements could be correctly established while inadequate realization capability remained insufficiently visible.

The Factor is distinct from Requirements Realization because it concerns the means of realization rather than the definition of what must be realized.

It is also distinct from Quality-State Determination because possessing realization capability does not establish that management can adequately determine the actual condition.

The specific forms of capability vary substantially across contexts.

A technical system, professional service, construction activity, support service, and information process may require very different capabilities.

The functional need to establish and maintain sufficient capability nevertheless remains.

Realization Capability is therefore justified as a separate Core Quality Factor.

6.3 Factor 3 - Quality-State Determination

The Core purpose of Quality-State Determination is:

“The degree to which management can determine the actual condition relevant to its responsibilities with sufficient scope, method, competence, Evidence use, criteria, interpretation, and current validity.”

Management System Function Realization requires more than requirements and realization capability.

Management must also be able to determine the actual condition relevant to its responsibilities.

Without adequate determination, management may continue operating without knowing whether requirements are being realized, whether an adverse condition exists, whether a change has affected realization, or whether previous information remains current.

Quality-State Determination may involve observation, inspection, testing, measurement, assessment, review, professional judgment, or other applicable determination methods.

The specific method is contextual.

The functional need is not.

The Factor must preserve visibility of determination scope, method, coverage, assessment capability, competence, evidentiary basis, applicable Quality Outcome Criteria, interpretation, Quality-State conclusion, and current validity.

Its distinct determination question is:

Can management adequately determine the actual condition relevant to its responsibilities?

If this Factor were removed, a Management System could have clear requirements and strong realization capability while remaining unable to establish the actual condition with sufficient confidence.

Quality-State Determination is also distinct from Evidence Quality.

Within WQI_QMS1, this Factor examines the management function used to determine actual condition.

Evidence Quality separately concerns the quality of the Evidence used to support Evidence-Supported Findings.

A well-designed determination function can still rely on insufficient-quality Evidence in a particular case, while high-quality Evidence does not by itself establish that the Management System's determination function is adequate.

Preserving this distinction prevents Evidence Quality from being substituted for Management System Quality.

Quality-State Determination is therefore justified as a separate Core Quality Factor.

6.4 Factor 4 - Quality-State Information

The Core purpose of Quality-State Information is:

“The degree to which Quality-relevant information remains sufficiently identifiable, connected, current, available, communicated, and protected for Management System Function Realization.”

Management depends on information, but its existence alone does not establish that it can adequately support Management System Function Realization.

A correct determination can fail to reach the responsible function.

A record can exist without a reliable connection to the relevant subject or requirement.

Information can remain in a system after it is no longer current.

Required information can be unavailable at an Interface or during a critical decision.

Information can be altered, lost, improperly used, or insufficiently protected.

Quality-State Information therefore asks:

Does Quality-relevant information remain sufficiently usable and connected for Management System Function Realization?

If this Factor were removed, the architecture could identify requirements, capability, and actual condition while failing to show whether the information needed to connect those functions remains usable.

The Factor is distinct from Quality-State Determination.

Determination concerns establishing the actual condition.

Information concerns maintaining and communicating Quality-relevant information so that management functions can use it.

The Factor is broader than recordkeeping; it concerns the functional usability of Quality-relevant information throughout Management System Function Realization.

Different contexts may use paper records, digital systems, direct communication, automated data exchange, professional reports, or other methods.

No particular information system is prescribed by the Factor.

Quality-State Information is therefore justified as a separate Core Quality Factor.

6.5 Factor 5 - Quality-State Influence

The Core purpose of Quality-State Influence is:

“The degree to which management can appropriately protect, contain, stabilize, respond, intervene, establish an acceptable resulting condition, address affected scope, and address cause and recurrence where required.”

Management System Function Realization requires the ability to act when management action is necessary.

A Management System may know that an adverse or unacceptable condition exists and may communicate that information correctly while still being unable to appropriately influence the condition.

The required influence may include immediate protection, containment, stabilization, response or disposition, intervention authority, timely action, establishment of an acceptable resulting condition, identification of affected scope, and attention to cause and recurrence where required.

The Factor therefore asks:

Can management appropriately influence the condition when management action is required?

If this Factor were removed, the architecture could show that management knows what is required, possesses realization capability, determines an adverse condition, and communicates it, while failing to reveal that management cannot adequately respond.

Quality-State Influence is distinct from Quality-State Determination because knowing the condition is not the same as influencing it.

It is distinct from Quality-State Information because communicating a required action does not establish the authority, capability, or timely intervention needed to perform it.

The form and extent of influence will vary by Context and by the applicable responsibility and authority of the Management System.

The Factor does not imply unlimited management control.

It examines the management influence that is applicable within the defined responsibility and authority.

Quality-State Influence is therefore justified as a separate Core Quality Factor.

6.6 Factor 6 - Lifecycle and Interface Realization

The Core purpose of Lifecycle and Interface Realization is:

“The degree to which Management System Function Realization remains sufficiently connected through relevant Lifecycle stages, changes, transfers, handoffs, and Interfaces.”

Management System Function Realization often extends across distributed functions, responsibilities, stages, systems, providers, or organizations.

Requirements, responsibilities, information, decisions, realization activities, determinations, and management actions may pass through different Lifecycle stages, organizational functions, providers, systems, or organizations.

Each local function may appear adequate while the connection between functions fails.

A requirement can be lost during transfer.

Responsibility can become unclear at an Interface.

A Quality State can be inadequately preserved or confirmed after a handoff.

A change can affect a downstream function without the effect being adequately connected.

The Factor therefore asks:

Does Management System Function Realization remain sufficiently connected across the relevant Lifecycle stages, changes, transfers, handoffs, and Interfaces?

If this Factor were removed, local adequacy within the other five Factors could conceal failures occurring between stages or responsibilities.

The other Factors can contribute to Interface performance, but none individually establishes whether the necessary functional continuity is preserved.

Lifecycle and Interface Realization is therefore not simply a general coordination category.

It addresses a distinct failure mode: loss of necessary connection across the structure through which Function Realization occurs.

The specific Lifecycle and Interface arrangements depend on Context.

Where such dependencies exist, however, the functional need remains.

Lifecycle and Interface Realization is therefore justified as a separate Core Quality Factor.

7. COLLECTIVE TESTING

7.1 Completeness Test

The six Factors must collectively provide sufficient coverage of the Management System Intended Function.

The Intended Function requires management to ensure, within its applicable responsibility and authority, that the required product, service, work, or other Intended Result is realized as required.

The six Factors cover the major functional questions necessary to examine this task: what must be realized; whether sufficient capability exists; whether the actual condition can be determined; whether Quality-relevant information remains usable; whether management can appropriately influence conditions requiring action; and whether these functions remain connected across relevant Lifecycle stages and Interfaces.

Many additional management subjects can be important, including leadership, governance, planning, resources, competence, communication, risk, documentation, monitoring, change, corrective action, and coordination.

Their importance does not automatically require separate Core Factor status.

Competence, for example, contributes to capability and determination.

Communication contributes to Quality-State Information.

Change can affect requirements, capability, determination, information, influence, and Lifecycle or Interface realization.

Responsibility and authority can affect several Factors.

At the present Core Methodological Resolution, no additional major functional area has been identified that requires separate Factor status in order to preserve necessary determination capability.

The six Factors are therefore sufficiently complete for the present Core architecture.

7.2 Distinctness and Overlap Test

Completeness alone is insufficient; the Factors must also remain distinguishable without creating unnecessary duplication.

Requirements Realization and Realization Capability interact, but one defines what must be realized while the other concerns the capability to realize it.

Realization Capability and Quality-State Determination interact, but the ability to realize a requirement is different from the ability to determine the actual condition.

Quality-State Determination and Quality-State Information interact, but establishing a condition is different from maintaining usable information about it.

Quality-State Information and Quality-State Influence interact, but knowing and communicating a condition is different from appropriately acting on it.

Lifecycle and Interface Realization interacts with every other Factor, but its distinct purpose is to preserve necessary functional connections across stages, changes, transfers, handoffs, and Interfaces.

Some Evidence may be relevant to more than one Factor.

The same event or condition may also affect multiple Factors.

This does not make the Factors redundant.

The Evidence-Supported Finding must remain connected to the specific management function being examined.

The overlap test therefore supports an architecture of interacting but distinguishable Factors. Overlap is acceptable where it reflects a real functional relationship; duplication would arise only if two Factors asked materially the same determination question at the same Methodological Resolution.

7.3 Removal Test

The removal test examines whether each Factor preserves determination capability that cannot be adequately retained by the other five.

Removing Requirements Realization creates a blind area concerning what management establishes and maintains as required.

Removing Realization Capability creates a blind area concerning whether management possesses and maintains the means needed for realization.

Removing Quality-State Determination creates a blind area concerning whether management can adequately determine the actual condition.

Removing Quality-State Information creates a blind area concerning whether Quality-relevant information remains usable and connected.

Removing Quality-State Influence creates a blind area concerning whether management can appropriately act when influence is required.

Removing Lifecycle and Interface Realization creates a blind area concerning whether necessary management functions remain connected across stages and Interfaces.

Each removal therefore eliminates a distinct determination capability.

The test supports retention of all six Factors.

It also demonstrates why the Factors should not be treated merely as interchangeable components of a general management score.

Each preserves different information about Management System Function Realization.

7.4 Non-Compensation and Failure-Visibility Test

The Factor architecture must prevent strong performance in one functional area from automatically concealing weakness in another.

This expresses non-compensation at the Factor level.

Strong Realization Capability does not automatically compensate for inadequate Requirements Realization.

Accurate Quality-State Determination does not compensate for unusable Quality-State Information.

Good information does not compensate for lack of appropriate Quality-State Influence.

Strong local performance does not compensate for loss of necessary continuity across an Interface.

Non-compensation does not mean that every weakness has the same significance.

The importance of a Finding depends on the applicable Quality Outcome Criteria, Context, criticality, relationships, and the Management System Intended Function and Intended Result.

The purpose is to preserve failure visibility.

Two Management Systems can produce the same broad result while having materially different underlying management conditions.

The Factor architecture should allow those differences to remain visible rather than being prematurely collapsed into an overall interpretation.

Evidence-Supported Findings relevant to each Factor should therefore be preserved before broader Management System Quality interpretation is made.

7.5 Consistency of Methodological Resolution

The six Factors must operate at a comparable functional level.

A Core architecture would become unstable if some Factors represented broad management functions while others represented individual procedures, records, roles, or detailed controls.

The six Factors each represent a major part of Management System Function Realization.

More detailed determination functions are addressed through the Core Quality Indicators beneath the Factors.

Context-specific expectations are addressed through applicable Quality Outcome Criteria.

Evidence supports the resulting Findings.

The hierarchy therefore remains:

Management System Function Realization

↓

Core Quality Factors

↓

Core Quality Indicators

↓

Applicable Quality Outcome Criteria

↓

Evidence

↓

Evidence-Supported Findings

If a proposed subject is more appropriately treated as an Indicator, Quality Outcome Criterion, Evidence item, or lower-level application issue, it should not be elevated to Core Factor status merely because it is important.

The six Factors remain at a sufficiently comparable Methodological Resolution to function as one Core architecture.

7.6 Collective Finding

The collective tests support treatment of the six Factors as one connected architecture at the present Core Methodological Resolution.

Together, the Factors preserve the major functional distinctions needed to examine Management System Function Realization without collapsing those distinctions into a single overall management judgment.

The collective analysis also shows that interaction among the Factors does not make them interchangeable. Each Factor retains a distinguishable determination purpose while contributing to the same Management System Intended Function.

Detailed management subjects remain examinable beneath the Factor level through Core Quality Indicators, applicable Quality Outcome Criteria, Evidence, and context-specific methods.

The collective architecture is therefore sufficiently coherent, functionally connected, and consistent in Methodological Resolution to proceed to cross-domain stress testing.

The separate question of whether the architecture should contain fewer or more than six Core Factors is addressed in Section 9.

8. CROSS-DOMAIN STRESS TESTING

8.1 Purpose of Cross-Domain Stress Testing

A Core Management System Quality architecture should not depend on one industry, profession, organizational form, or type of work.

The six Factors should remain meaningful when management is used to realize materially different types of results.

These can include physical, technical, professional, support, project, maintenance, information, and other results, realized within one organization or across multiple organizations.

Cross-domain stress testing therefore asks whether the six functional areas remain recognizable and distinguishable as the Context changes.

Requirements Realization should remain meaningful even though the substantive requirements differ.

Realization Capability should remain meaningful even though the required methods, competence, equipment, infrastructure, or providers differ.

Quality-State Determination should remain meaningful even though determination methods and Evidence differ.

Quality-State Information should remain meaningful even though information systems and communication methods differ.

Quality-State Influence should remain meaningful even though the available forms of management action differ.

Lifecycle and Interface Realization should remain meaningful even though Lifecycle structures and organizational relationships differ.

The test concerns functional stability, not uniform implementation.

8.2 Different Types of Managed Results

Management Systems can be used to realize very different Intended Results.

Examples include a physical product, a technical operation, construction or installation work, maintenance or repair, a professional service, human support, an administrative result, or an information service.

Some results are primarily physical.

Others depend heavily on professional judgment, human interaction, information, or continuing service.

Some are realized within a short period.

Others extend through long Lifecycles.

Despite these differences, management still needs to establish what is required, maintain sufficient capability, determine relevant actual conditions, maintain usable Quality-relevant information,

appropriately influence conditions where required, and preserve necessary connections through relevant stages and Interfaces.

The substantive application changes.

The six underlying functional questions do not need to change.

Variation can therefore be addressed through applicable Indicators, Quality Outcome Criteria, Evidence, methods, Boundaries, Interfaces, and Context without changing the Core Factor architecture.

8.3 Different Realization Methods and Technologies

The methods used to realize an Intended Result can range from primarily manual or human activity to machinery, automation, digital systems, software, remote operation, or highly specialized technology.

Technological change can substantially alter how management functions are performed.

It does not necessarily alter the functions that need to be examined.

An automated process still requires applicable requirements.

Its realization capability still needs to be adequate.

Its actual condition still needs to be determinable.

Quality-relevant information still needs to remain usable and connected.

Management still requires appropriate influence where intervention is necessary.

Lifecycle and Interface connections still need to be preserved.

Technology can change the methods, Evidence, Quality Outcome Criteria, and detailed determination questions.

The cross-domain test does not identify a need to change the six Core Factors merely because realization technology changes.

8.4 Different Organizational and Provider Arrangements

Management System Function Realization may occur within one organization or through combinations of internal functions, contractors, suppliers, external providers, partners, or other organizations.

As realization becomes distributed, responsibilities, authority, information flows, requirement transfers, and Interfaces can become more complex.

The six Factors remain applicable.

Requirements still need to be established and connected.

Provider and organizational capability must still support realization within the applicable management responsibility.

Relevant conditions must still be determinable.

Quality-relevant information must remain usable across the functions that depend on it.

Management influence must remain available where responsibility and authority require action.

Lifecycle and Interface Realization becomes particularly important because requirement transfer, Quality-State continuity, responsibility, and downstream confirmation can fail between otherwise capable organizations.

These variations change the applicable Boundaries, Interfaces, responsibility, authority, and Evidence.

They do not require a different Core Factor architecture.

8.5 Different Lifecycle Conditions and Change

Management System Function Realization can occur across planning, design, development, procurement, realization, operation, maintenance, modification, transfer, suspension, restart, closure, or other applicable Lifecycle conditions.

Requirements, capability, responsibilities, information, and actual conditions can change over time.

The six Factors remain meaningful through these changes.

Requirements Realization addresses whether applicable requirements remain correctly established and maintained.

Realization Capability addresses whether capability remains adequate, including after change.

Quality-State Determination addresses whether the actual condition can be determined with current validity.

Quality-State Information addresses whether relevant information remains connected and current.

Quality-State Influence addresses whether management can appropriately respond when changed conditions require action.

Lifecycle and Interface Realization addresses whether necessary continuity is preserved through transitions, transfers, and changes.

Not every detailed Indicator will necessarily be applicable in every Lifecycle condition.

Applicability depends on the functions and dependencies that exist.

The Core Factor architecture itself does not require change merely because the Lifecycle condition changes.

8.6 Normal, Unusual, and Degraded Conditions

Management Systems may operate under normal, unusual, abnormal, emergency, degraded, disrupted, or other non-routine conditions.

Such conditions can change the requirements, available capability, information, determination methods, intervention options, responsibilities, or Interfaces relevant to Function Realization.

The six Factors remain meaningful.

Management still needs to understand applicable requirements.

Capability must remain adequate for the applicable conditions.

Relevant actual conditions must be determinable.

Quality-relevant information must remain usable.

Management must be able to appropriately influence conditions within its responsibility and authority.

Necessary Lifecycle and Interface connections must remain sufficiently preserved.

This does not mean that every Management System must be capable of addressing every conceivable condition.

Applicable conditions are established in relation to the Quality Object, requirements, responsibility, authority, Boundaries, Interfaces, Context, and Intended Result.

The stress test asks whether the six Factors remain capable of examining Management System Function Realization when those conditions change.

They do.

8.7 Cross-Domain Stress-Test Finding

The cross-domain stress test examined variation in the type of managed result, realization method and technology, organizational and provider arrangement, Lifecycle condition and change, and normal, unusual, and degraded operating conditions.

Across these variations, the detailed application of Management System Quality Determination changes.

The underlying six functional needs remain recognizable.

No additional Core management function has been identified solely because the domain changes.

No existing Factor becomes unnecessary solely because the domain changes.

No combination of Factors becomes justified merely because a particular application uses the same activity or Evidence across several functions.

Variation is therefore primarily absorbed beneath the Factor level through applicable Core Quality Indicators, Quality Outcome Criteria, Evidence, determination methods, Boundaries, Interfaces, and Context.

Core invariance does not mean uniform implementation; it means that the functional architecture remains stable while application adapts.

The cross-domain stress test therefore supports retention of the six-Factor architecture at the present Core Methodological Resolution.

This is a bounded methodological finding, not a statistical or empirical proof that the architecture is permanently complete.

9. WHY SIX, AND NOT FEWER OR MORE

9.1 Why Fewer Factors Would Reduce Determination Capability

A smaller number of Core Factors would be preferable only if the same determination capability and failure visibility could be preserved.

The analysis does not support that conclusion.

Combining Requirements Realization with Realization Capability would obscure the difference between knowing what must be realized and possessing the means to realize it.

Combining Realization Capability with Quality-State Determination would obscure the difference between the ability to realize a requirement and the ability to determine the actual condition.

Combining Quality-State Determination with Quality-State Information would obscure the difference between establishing a condition and maintaining usable information about it.

Combining Quality-State Information with Quality-State Influence would obscure the difference between knowing and communicating a condition and being able to appropriately act on it.

Absorbing Lifecycle and Interface Realization into the other Factors would reduce visibility of failures occurring between otherwise adequate stages, functions, responsibilities, or organizations.

The fact that Factors interact does not justify combining them.

Their interaction is part of Management System Function Realization, while their separation preserves distinct determination questions.

Reducing the architecture below six Factors would therefore reduce determination capability at the present Core Methodological Resolution.

9.2 Why Additional Factors Are Not Required at the Present Core Methodological Resolution

The opposite question is whether Management System Quality requires more than six Core Factors.

Many additional management subjects can be identified.

Examples include leadership, governance, planning, risk, competence, resources, communication, documentation, monitoring, change, corrective action, coordination, learning, and improvement.

These subjects can be important.

Importance alone, however, does not establish the need for a separate Core Factor.

Leadership and governance can affect requirements, capability, information, influence, responsibilities, and Interfaces.

Planning can contribute across all six Factors.

Competence and resources are relevant to Realization Capability and, where applicable, Quality-State Determination.

Communication is directly relevant to Quality-State Information and Lifecycle or Interface realization.

Monitoring, inspection, assessment, and review contribute to Quality-State Determination.

Corrective action and intervention contribute to Quality-State Influence.

Change can affect every Factor and should be examined where its effects occur.

Risk can influence requirements, capability, determination, information, influence, and Lifecycle or Interface decisions. Treating risk as a separate Factor could detach it from the functions through which risk is actually managed.

Learning and improvement can result from Findings, changes, and management actions across the Factor architecture.

Creating additional Core Factors for these subjects could therefore mix different Methodological Resolutions, duplicate existing functions, or move the Core toward a detailed management checklist.

The architecture is not permanently closed to additional Factors.

A new Factor would be justified if a substantive management function were identified that is not adequately represented by the existing Factors, whose omission creates a meaningful determination blind area, that is distinguishable and Evidence-determinable, and that remains sufficiently stable across contexts.

At the present Core Methodological Resolution, no such additional functional area has been identified.

More than six Core Factors are therefore not presently required.

9.3 Why Six Is the Supported Core Resolution

The question of why the architecture contains six Factors cannot be answered simply by counting important management subjects.

The number results from the Methodological Resolution at which Management System Function Realization is examined.

At a broader resolution, several of the six Factors could be combined.

Management System Quality might then be described through only a few general categories such as requirements, capability, and control.

Such an architecture would be simpler, but it would hide important differences among determining an actual condition, maintaining Quality-relevant information, influencing a condition requiring action, and preserving necessary connections across Lifecycle stages and Interfaces.

At a finer resolution, each of the six Factors could be divided into numerous separate functions.

Personnel competence, provider capability, equipment capability, communication, intervention authority, change, requirement transfer, downstream confirmation, and many other subjects could each be treated as separate categories.

Such an architecture would provide more detail, but the Core Factors would begin to operate at the level intended for Core Quality Indicators or context-specific application.

The supported resolution lies between these two extremes.

At this resolution, six functional areas remain separately visible:

Requirements Realization

Realization Capability

Quality-State Determination

Quality-State Information

Quality-State Influence

Lifecycle and Interface Realization

Each Factor represents a necessary and distinguishable part of Management System Function Realization.

Removing a Factor creates a meaningful determination blind area.

Combining Factors reduces visibility of materially different management functions.

Adding further Factors is not presently necessary because the additional subjects examined can be represented within the six Factors or at a more detailed level of the architecture.

The number six therefore has no independent methodological significance.

It is the result of applying the selected Core Methodological Resolution to the Management System Quality Determination task.

If the Methodological Resolution were intentionally changed, the number of analytical categories could also change.

For WQI_QMS1, however, the purpose is to establish a stable Core architecture that is sufficiently detailed to preserve important functional distinctions while remaining sufficiently general for cross-context application.

Under the methodology and testing presented in this paper, six Factors provide that balance.

The six-Factor architecture is therefore supported at the present Core Methodological Resolution.

10. STABILITY AND CONTROLLED REVISION

10.1 Stability of the Core Factor Architecture

A Core architecture must be capable of development, but it should not change whenever a new management practice, technology, application, or terminology appears.

The six Core Quality Factors are intended to provide the stable functional structure of the Management System Quality Architecture at the present Core Methodological Resolution.

Their stability comes from the level at which they are defined.

The Factors are not based on a particular organizational structure, industry, technology, management method, or set of procedures.

They are based on the major functional areas that must remain visible when examining whether a Management System realizes its Intended Function.

Requirements may change, but the need for Requirements Realization remains.

Methods, technologies, personnel, providers, and equipment may change, but the need for Realization Capability remains.

Determination methods and Evidence may change, but the need for Quality-State Determination remains.

Information technologies and communication methods may change, but the need for Quality-State Information remains.

Forms of intervention may change, but the need for Quality-State Influence remains.

Lifecycle structures and organizational relationships may change, but the need to preserve necessary Lifecycle and Interface connections remains where those dependencies exist.

This functional basis allows the detailed application of the architecture to evolve without automatically requiring changes to the Core Factors.

New management practices can therefore be examined through the existing architecture before a new Factor is considered.

Similarly, a new technology, organizational arrangement, or type of managed result should not by itself justify changing the Core.

The first question should be whether the new condition creates a management function that cannot be adequately represented through the existing six Factors.

If it does not, the variation should normally be addressed through the applicable Core Quality Indicators, Quality Outcome Criteria, Evidence, Context, or other context-specific elements of the Quality Determination.

Stability does not mean that the architecture is permanently fixed; revision should require a substantive methodological reason rather than ordinary variation in application.

This approach preserves comparability and continuity in Management System Quality Determination while allowing the methodology to develop when justified.

10.2 Conditions That Could Justify Revision

Core stability does not preclude revision.

Revision should remain possible when substantive methodological evidence shows that the existing architecture no longer provides an adequate basis for Management System Quality Determination.

A revision may be justified if analysis or application identifies a major part of Management System Function Realization that is not adequately represented by the existing six Factors.

Such a finding would need to show more than the importance of a newly recognized management subject.

The proposed function would need to be necessary, distinguishable from the existing Factors, relevant at the Core Methodological Resolution, meaningfully determinable through Evidence, and sufficiently stable across different management contexts.

Revision may also be justified if two existing Factors are shown not to preserve genuinely different determination questions.

If their separation creates unnecessary duplication and they can be combined without losing important failure visibility or determination capability, consolidation may be appropriate.

A further reason for revision could arise if a Factor proves to contain two or more fundamentally different functions that cannot be adequately examined together at the same Core Methodological Resolution.

In that case, separation may need to be considered.

Repeated application may also reveal previously unrecognized Boundaries, Interfaces, failure modes, or functional relationships that challenge the existing architecture.

Such findings should be examined methodologically rather than treated automatically as reasons for structural change.

A proposed revision should therefore answer several questions:

What determination capability is currently missing or unnecessarily duplicated?

Why cannot the issue be adequately addressed through the existing Factors and Core Quality Indicators?

Would the proposed change improve failure visibility or functional distinctness?

Would the revised architecture remain coherent at the Core Methodological Resolution?

Would the change remain meaningful across materially different management contexts?

What Evidence and methodological analysis support the proposed revision?

Revision should occur only when these questions demonstrate a substantive improvement to the Core architecture.

This threshold helps distinguish genuine methodological development from changes that can be addressed through context-specific application, clarification, or refinement at a lower level of the architecture.

10.3 Controlled Revision Principle

Any revision of the six-Factor Core architecture should be controlled, traceable, and methodologically justified.

The purpose of controlled revision is to allow the architecture to develop without weakening its stability or changing its meaning through informal additions, deletions, or renaming.

A proposed revision should begin with a clearly identified methodological issue.

The issue should describe the limitation observed in the existing architecture, the part of Management System Quality Determination affected, and the reason the existing Core structure may no longer be sufficient.

The proposed change should then be tested using the same principles applied in deriving the current architecture.

These include:

necessity;

removal consequence;

distinctness;

non-redundancy;

failure visibility;

Evidence and determinability;

cross-context applicability; and

consistency of Methodological Resolution.

The proposed change should also be tested against the architecture as a whole.

Adding, removing, combining, dividing, or materially redefining a Factor can affect the boundaries and interpretation of other Factors.

Revision should therefore consider the resulting six-Factor or revised architecture collectively rather than examining only the proposed change in isolation.

The basis for the revision should be documented.

This should make clear what changed, why it changed, what methodological analysis supported the change, and how the revised architecture relates to the preceding version.

Ordinary differences in application should not be treated as revisions of the Core architecture.

Different industries, organizations, technologies, Lifecycle conditions, Quality Outcome Criteria, Evidence, or assessment methods can normally be addressed within the existing Core structure.

Controlled revision is appropriate only when the Core functional architecture itself requires change.

This principle allows the Management System Quality architecture to remain stable enough for consistent use while remaining open to future methodological development where that development is sufficiently justified.

10.4 Status of the Present Architecture

The six-Factor architecture represents the presently supported Management System Quality Architecture at the Core Methodological Resolution.

The analysis in this paper derives the Factors from the Management System Intended Function and Intended Result and tests them through functional failure modes, necessity, removal consequence, distinctness, non-redundancy, Evidence and determinability, cross-context applicability, collective completeness, non-compensation, and consistency of Methodological Resolution.

The analysis supports six Core Quality Factors:

Requirements Realization

Realization Capability

Quality-State Determination

Quality-State Information

Quality-State Influence

Lifecycle and Interface Realization

At the present Core Methodological Resolution, each Factor preserves a necessary and distinguishable part of Management System Function Realization.

The analysis does not identify a Factor that can be removed or combined with another without reducing determination capability or failure visibility.

It also does not identify an additional major functional area that presently requires separation as another Core Quality Factor.

This status should not be interpreted as a claim of permanent or empirically proven completeness.

The architecture remains open to controlled revision if future methodological analysis, application, or Evidence identifies a substantive reason for change.

Until such a reason is established, the six Factors provide the stable Core architecture for Management System Quality Determination under WQI_QMS1.

Detailed resolution beneath these Factors is provided through the Core Quality Indicators.

The methodological justification of those Indicators is addressed separately in the WQI Management System Quality Indicators Justification.

11. CONCLUSION

This paper examined why Management System Quality Determination resolves, at the present Core Methodological Resolution, into six Core Quality Factors.

The analysis did not begin by assuming that six Factors were required.

It began with the Management System Intended Function:

“To ensure, within the applicable responsibility and authority of the Management System, that the required product, service, work, or other Intended Result is realized as required.”

From this function, the analysis identified the major functional needs that must remain visible when Management System Quality is determined.

Management must establish and maintain what is required.

Management must establish and maintain the capability needed to realize what is required.

Management must be able to determine the actual condition relevant to its responsibilities.

Quality-relevant information must remain sufficiently usable for the management functions that depend on it.

Management must be able to appropriately influence conditions when management action is required.

These management functions must remain sufficiently connected through relevant Lifecycle stages, changes, transfers, handoffs, and Interfaces.

These functional needs resolve into six Core Quality Factors:

Requirements Realization

Realization Capability

Quality-State Determination

Quality-State Information

Quality-State Influence

Lifecycle and Interface Realization

Each Factor was examined for necessity, removal consequence, distinctness, non-redundancy, Evidence and determinability, and cross-context applicability.

The Factors were then tested collectively for completeness, overlap, non-compensation, failure visibility, and consistency of Methodological Resolution.

Cross-domain stress testing further examined whether the architecture remains meaningful when the type of managed result, realization method, technology, organizational arrangement, Lifecycle condition, Interface structure, and operating condition change.

The methodological analysis supports the six-Factor architecture at the present Core Methodological Resolution.

Removing a Factor creates a meaningful blind area in Management System Quality Determination.

Combining Factors reduces visibility of distinguishable management functions.

Adding further Core Factors is not presently necessary because the additional management subjects examined can be represented within the existing Factors or at a more detailed level of the architecture.

The conclusion is deliberately bounded.

This paper does not claim statistical or empirical proof that six Factors are permanently complete or universally optimal.

It provides a methodological justification for the presently supported WQI Core architecture.

At the present Core Methodological Resolution, the six Factors provide a sufficiently complete, distinct, non-redundant, functionally connected, and cross-context structure for determining Management System Quality.

The architecture should remain stable unless further methodological analysis, application, or Evidence establishes a substantive reason for controlled revision.

The next level of resolution is provided by the Core Quality Indicators.

Their separate methodological justification addresses why the individual Indicators are necessary within the six-Factor architecture and why they collectively provide the presently supported Indicator structure of WQI_QMS1.

Copyright and Use Notice

© 2026 Whole Quality Institute (WQI). All rights reserved.

This document is a WQI methodological article prepared for development, review, research, education, and standards-development purposes. No part of this document may be represented as an independently issued WQI standard unless formally designated and published as such by WQI.

References to WQI, WQI_VOC1, WQI_QMS1, Management System Quality, Quality Factors, Quality Indicators, and related Whole Quality terminology are used within the WQI methodological framework. The analysis in this article explains and stress-tests the architecture; it does not establish legal, regulatory, engineering, certification, contractual, or organization-specific management requirements.

This article does not replace or supersede applicable laws, regulations, technical standards, contracts, organizational requirements, or other applicable Reference Layer sources.

Where this article is revised, the revision should preserve Traceability to WQI_QMS1, WQI_VOC1, and the methodological basis used to derive and test the six-Factor Management System Quality Architecture.