



# Competency Models for Evaluating the Proficiency of Solution Architects in International IT Companies

Mykhailo Krasovskyi

Lead Solution Architect at Trinetix, Kyiv, Ukraine.

## Abstract

*The article examines competency models for evaluating the proficiency of solution architects in international IT companies, focusing on career levels, skill categories, scoring logic, and the practical use of assessment matrices. The aim of the study is to identify the structural principles of existing architectural competency frameworks and to demonstrate how a weighted, multidimensional matrix can support hiring, promotion, development, and project allocation. The research's relevance lies in the expanding role of the architect across cloud-native architectures, microservices, enterprise transformation, compliance demands, and generative AI. The novelty lies in combining skill weighting, criticality tiers, numeric ratings, and aggregate thresholds into a single operational model. The main findings show that such a matrix creates a shared assessment language, improves consistency in staffing and promotion decisions, clarifies growth trajectories, supports internal mobility, and reduces dependence on individual expert judgment. The article will be useful for IT managers, architecture practice leads, HR specialists, software architects, solution architects, and researchers studying professional competency assessment.*

**Keywords:** Software Architect, Solution Architect, Competency Model, Skill Matrix, Career Ladder, IT Talent Management, Architecture Practice.

## INTRODUCTION

The role of the solution architect has been reshaped by two parallel forces, and in international IT consultancies, the related solution architect role has followed the same trajectory. The first force is the industry-wide shift from monolithic on-premises systems to cloud-native and microservices architectures, which has changed the technical decision space in which architects operate. Work affirms the transition, as architects need to reason about distributed systems, observability, platform engineering, and new integration patterns with generative AI components (Martínez-Fernández et al., 2022). The second is the transition of additional non-technical tasks onto the architect, such as pre-sales, transformation advisory, mentoring, and program management (Tavoletti et al., 2021). The modern architect operates at the intersection of engineering, business, and leadership, and this hybridity is now treated as a defining feature of the role at every level. Senior practitioners no longer hold a monopoly on it.

The hybrid nature of the role creates a measurement problem. A typical software engineering career ladder uses a small number of ordinal levels, such as junior, mid, senior,

and staff, anchored by coding scope and autonomy. For architects, this is insufficient on two counts. First, the role's categorical breadth means that two architects at the same nominal level may have very different strengths, with one strong in pattern-level design and the other in stakeholder management and pre-sales work. Flat ladders cannot represent this difference without collapsing the distinctions that matter most (Assyne et al., 2022). Second, when companies grow architectural practices beyond a handful of individuals, the absence of a formal model has downstream effects on hiring quality, retention, internal mobility, and the concentration of critical knowledge, often referred to as the bus factor problem (Jabrayilzade et al., 2022). Existing industry frameworks address parts of this gap, yet they are either too general to be operational at the level of a single company or too proprietary and tied to a single vendor's certification path (Bowers & Sabin, 2023).

This paper pursues three research questions. The first asks which competency models for software and solution architects are documented in the open literature and in major industry frameworks. The second asks which structural principles distinguish these models from one another, with candidates

**Citation:** Mykhailo Krasovskyi, "Competency Models for Evaluating the Proficiency of Solution Architects in International IT Companies", Universal Library of Engineering Technology, 2026; 3(2): 118-123. DOI: <https://doi.org/10.70315/uloap.ulete.2026.0302018>.

including the number of career levels, the way skills are organised into categories, the presence of skill weighting, the choice of scoring scale, and the use of aggregate thresholds for level confirmation. The third asks how a weighted multi-dimensional model can be operationalised in a mid-sized international IT company, and what effects on the architectural practice become observable after such a model is introduced. The contribution is twofold. The paper offers a typology of existing approaches and a structured description of one production model implemented in an international consultancy of around 50 architects.

## METHODS

This work combines a narrative review of existing competency models with a structured description of a practitioner case. It is not a systematic review in the PRISMA sense, and it does not present empirical metrics from a controlled deployment. The narrative approach was chosen because the relevant material is distributed across academic publications, professional body frameworks, and grey literature. The case description aims at illustrative generalisation, and statistical generalisation lies outside its scope (Wohlin, 2021).

The case is the architecture practice of Trinetix, an international IT company where the author serves as Architecture Practice Lead. The practice currently comprises around fifty architects working under shared standards for documentation, pre-sales, and delivery. The competency matrix presented here is the model used internally to assess architects and to inform promotion, development, and hiring decisions. This paper describes the structural design of the matrix, covering career levels, categorical blocks, the principles of the scoring system, and a generalised account of skill progression. Several elements have been deliberately omitted from the paper. These include the full table of skill descriptions, the exact numerical weights applied at each level, the precise minimum thresholds for level confirmation, individual assessment results, and any client-related context.

**Table 1.** Comparison of representative architect competency frameworks across seven structural dimensions.

Dimension	SWECOM	IASA BTABoK	SFIA	Bredemeyer	HR ladders
Number of levels	5	5 (per pillar)	7	4	3–6 (varies)
Architect focus	Software eng., incl. architect	Architect-specific	Broad IT, architect roles	Architect-specific	Org-specific
Categorical structure	Skill areas + skills	Five competency pillars	Categories × skills × levels	Five competency areas	Variable
Weighting per skill	No	No (qualitative)	No	No	Rare
Criticality tiering (Required / Important / Optional)	Implicit	Implicit	No	No	Rare
Numeric scoring scale	Level-based	Maturity (qualitative)	1–7 level	Level 1–4	Free-form
Aggregate level threshold	Per-skill criteria	No	Per-skill criteria	No	Rare
Openness	Open (IEEE-CS)	Open (IASA)	Subscription	Open	Proprietary

Two observations emerge from Table 1. First, no framework currently in use combines explicit skill weighting, criticality

The intent is to make the design principles transferable while preserving the specifics of the production instrument.

## RESULTS

### A Typology of Existing Competency Models

The reviewed models fall into two broad families. The first family consists of professional standards and bodies of knowledge, including SWECOM, IASA BTABoK, and SFIA. These are open or semi-open frameworks maintained by professional bodies, intended to apply across organisations and to support certification, curriculum design, and gap analysis. They share three properties: a multi-level structure of five to seven levels, a categorical partition of skills into areas, and an emphasis on industry-wide comparability. They differ in their granularity and in how directly they target the architect role. SWECOM treats the architect as a specialisation within a broader software engineering taxonomy. The IASA BTABoK is architect-centric and organises competencies into five pillars: business technology strategy, IT environment, quality attributes, design, and human dynamics. SFIA covers a wide range of IT roles with a uniform seven-level scale and is used by many large enterprises as a baseline.

The second family consists of author-driven and corporate models. The Bredemeyer framework was developed for and refined by working architects. It identifies five competency areas, which are technology, strategy, organisational politics, consulting, and leadership, and rates each across four levels of mastery. Corporate engineering ladders and career architectures, as documented in recent practitioner studies and academic case work, tend to be narrower (Tavoletti et al., 2021). They emphasise observable behaviours at a small number of levels and serve as HR instruments for promotion. Their weakness is that they flatten the architect role into a few behavioural cues and rarely distinguish between core and supporting competencies. Table 1 summarises the seven structural dimensions used to contrast these frameworks.

tiering, a numeric scoring scale, and an aggregate level threshold in a single integrated instrument. Most provide a

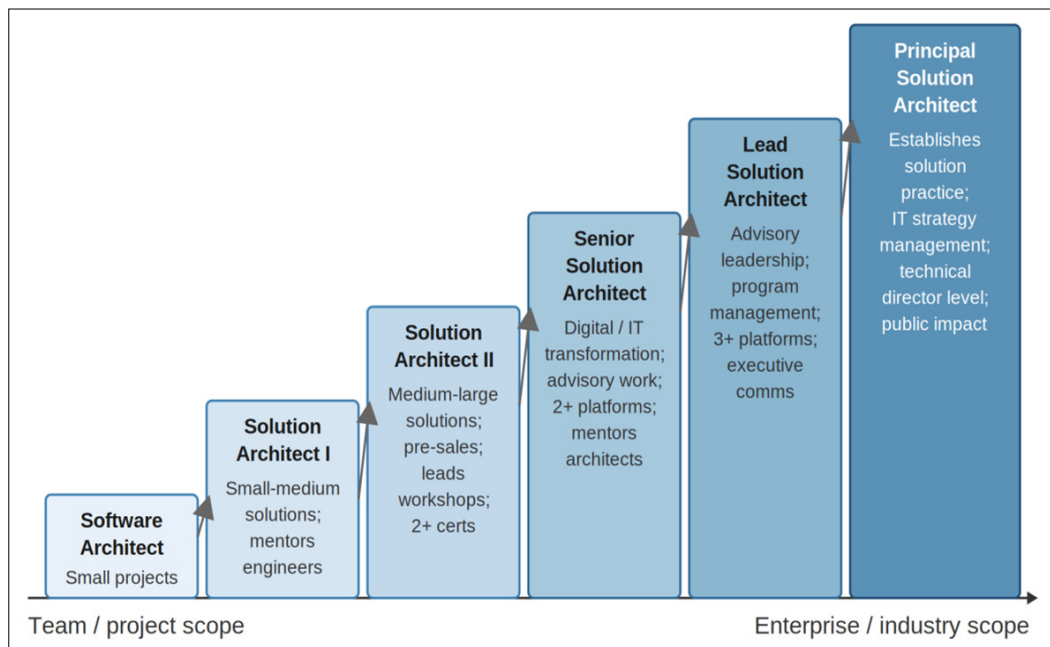
subset of these mechanisms. Second, the open frameworks tend toward breadth and qualitative depth, while corporate ladders tend toward operational simplicity at the cost of expressiveness. The space between these two poles is where the case described in Section 3.2 is positioned. Instruments that occupy this middle ground are operationally usable inside a single company and are still structurally rich enough to capture the hybridity of the architect role.

**A Weighted Multi-Dimensional Matrix: The Trinetix Case**

The Trinetix matrix is structured as a two-dimensional grid. Six career levels run along one axis, and six categorical blocks run along the other. Each cell in the grid contains level-specific behavioural descriptors for the skills in that block. A multi-dimensional scoring system, described in Section 3.2.3, is layered on top of this grid.

**Six Career Levels**

The career ladder progresses through six positions of widening scope. The entry point is Software Architect, with scope limited to small projects and team-leader responsibilities. Solution Architect I and Solution Architect II follow, owning solutions of growing size and leading pre-sales workshops. The Senior Solution Architect leads digital and IT transformation work and advisory engagements. Lead Solution Architect adds program management, multi-platform solutions, and tech-politics issues at executive level. The final level is Principal Solution Architect, who establishes solution practice at the organisational level, operates at technical director scope, and contributes to public discourse and industry-level achievements. The ladder is illustrated in Figure 1.



**Figure 1.** The six-level Solution Architect career ladder, with indicative scope at each level.

**Six Categorical Blocks**

The six categorical blocks group the skills assessed at each level. Professional Experience captures years of practice, project complexity, fluency in methodologies, and the architectural artefacts the architect has produced. Subject Matter Expertise tracks the number of subject areas in which the architect is recognised as an expert. Client Relations covers communication, presentation, whiteboarding, and the ability to lead client discussions of widening scale and seniority. Leadership and Teamwork combines team guidance, mentorship, interviewing, and senior leadership behaviours such as executive communication and acting as a last-resort escalation point for complex issues. Project Roles enumerates the positions the architect is qualified to occupy on engagements, from team lead to project supervisor and consultant. Self-development covers the architect’s own learning trajectory, including vendor certifications and awareness of industry trends.

**Skill Grouping**

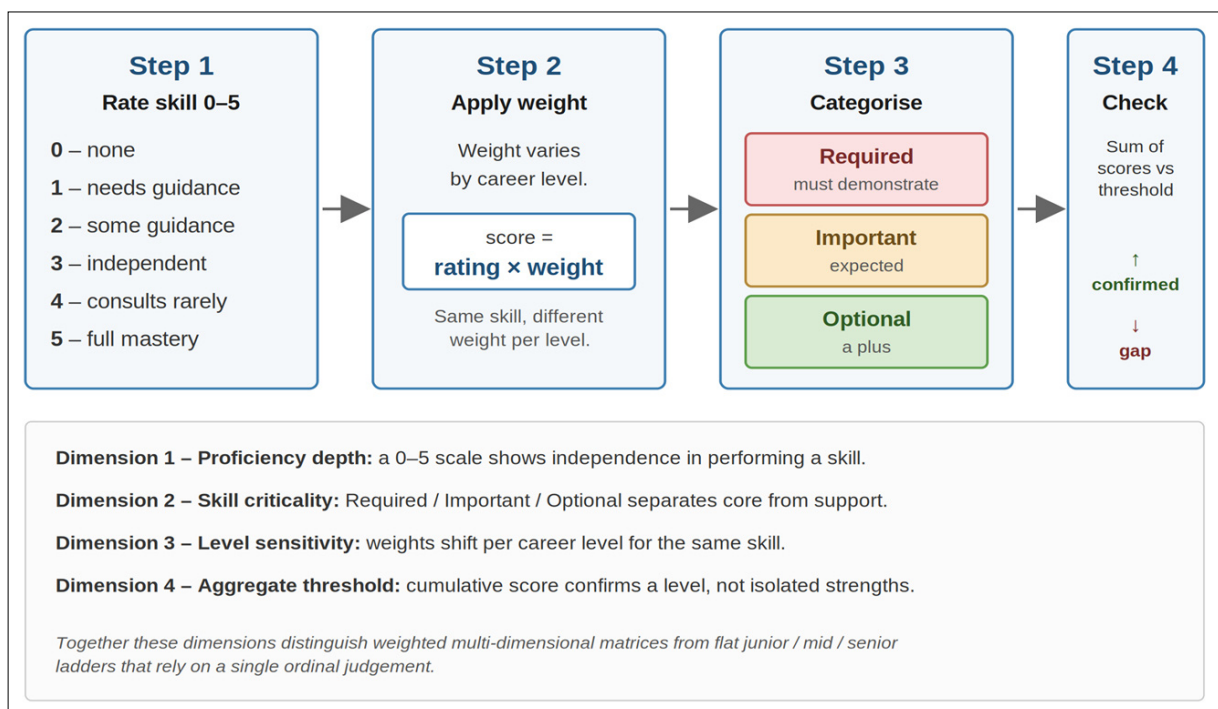
Inside the six categorical blocks, around thirty individual skills are tracked. They fall into four functional groupings. Technical expertise covers Software Development, Architecture Design, Application Architecture Patterns and Styles, Databases, and SDLC and methodologies. This grouping forms the technical foundation on which architectural decisions rest. Managerial competencies include Strategic Leadership, Project Management and Delivery, and Estimation and Work Planning, which together reflect the architect’s role in shaping how work is structured and delivered. Soft skills include Communication, English proficiency, Delivery, Change, and Innovation, Leadership and Mentorship, Teamwork and Collaboration, Ongoing Learning and Knowledge-sharing, and Flexibility, all of which address the human dimension of architectural work. Business and enterprise competencies include: Business Acumen and Consultancy; Business Needs and Goals; Business Value and Justification;

Business Strategy and Operating Model Innovation; Business Architecture; Stakeholder Management; Core Enterprise Architecture; Enterprise Architecture Body and Frameworks; Technology Consulting; and Program Management. These business competencies make the architect’s responsibility for connecting technical decisions to business value visible inside the matrix, a connection that remains implicit at best in many flat ladders.

### Principles of the Scoring System

The scoring system rests on four structural principles. The first is that each skill is rated on a 0 to 5 scale, where 0 indicates no experience, intermediate values capture practice that grows in independence and covers more of the described behavioural points, and 5 indicates full proficiency

without external guidance. The second is that each skill is then assigned a weight specific to the career level being assessed, so a skill that is foundational at a junior level may carry a heavier weight at a senior level, where its strategic dimension matters more. The third is that each skill is categorised as Required, Important, or Optional for the level under consideration. Required skills must be demonstrated to confirm the level. Important skills are expected and contribute a large share of the aggregate score, and Optional skills are recognised as a positive signal and do not block level confirmation. The fourth is that each career level has a defined minimum aggregate score, computed as the sum of ratings multiplied by weight across all skills, which an architect must reach for the level to be confirmed. Figure 2 illustrates these four steps schematically.



**Figure 2.** Conceptual schema of the weighted multi-dimensional scoring system used in the case matrix

### Illustrative Progression

To make the design concrete without reproducing the matrix in full, it helps to follow a single skill across the six levels. The chosen skill is Architecture Design. At the Software Architect level, the focus is on designing application architectures, applying design patterns, and documenting blueprints. At Solution Architect I and II levels, the architect leads application architecture design for complex solutions and ensures alignment with enterprise standards. At the Senior Solution Architect level, the work shifts toward application portfolio inventory, business-application mapping, and portfolio rationalisation. At the Lead Solution Architect level, the architect defines enterprise-wide design principles and approves architectural designs for enterprise-critical initiatives. At the Principal Solution Architect level, the same skill is exercised through setting design vision at the enterprise level and influencing industry practice. The

same skill is therefore assessed at every level, and each step raises both the weight and the demands of the behavioural anchors.

## DISCUSSION

### Where the Case Sits in the Typology

The case matrix aligns with established frameworks in its categorical structure. Technical, managerial, soft and business competencies are also reflected in the pillars and skill areas in the SWECOM, IASA BTABoK, SFIA and Bredemeyer frameworks, and are in line with more recent literature on the evolving role of the architect seeing stakeholder management, enterprise architecture and program management as first-class competencies at all levels, not just for senior practitioners (Lercher et al., 2024). Where the case goes beyond the reviewed frameworks is in combining all four scoring mechanisms in a single instrument. The

four mechanisms are numeric rating, level-specific weights, required, important, Important and Optional tiering, and an aggregate threshold. The reviewed open frameworks each provide a subset of these mechanisms, and none of them provides the full combination. The case, in turn, is more modest than the open frameworks in one respect. It does not aspire to be a universal body of knowledge. It is shaped by the operational context of an international consultancy with significant pre-sales activity, which is reflected in the weight given to client relations, business acumen, and technology consulting at every level.

### Observed Effects of Standardisation

The matrix was introduced as part of a broader standardisation initiative that covered documentation, pre-sales processes, and engineering practices. The effects of this initiative are best described qualitatively. Hiring decisions became more consistent across interviewers because the matrix supplied a shared vocabulary for what each level requires. Retention improved as architects gained a clearer picture of what was expected for the next promotion. Internal mobility increased because architects could be matched to engagements based on documented skill profiles, helping mitigate bus-factor risk. Time spent on pre-sales and architectural documentation decreased as standardised templates reduced duplicate effort. These effects are consistent with the broader literature on the operational impact of formal competency frameworks (Wohlin, 2021) and with practitioner accounts of mature architectural practices (Tavoletti et al., 2021). They are offered as qualitative observations from a single practice, and treating them as empirical metrics would overstate what the case can support.

### The Role Under Cloud-Native and Emerging-Technology Pressure

The architect's role has continued to evolve during the period the matrix has been in use. The shift to cloud-native and microservices architectures has elevated the importance of enterprise architecture, technology consulting, and compliance competencies, all of which are now first-class blocks in the matrix. The integration of generative AI into solution designs has introduced a new dimension that touches several existing skills at once, including architecture design, patterns and styles, and compliance assessment. The dimension is distributed across these skills, and it has not become a single skill of its own. This pattern of evolution carries a methodological consequence. Competency matrices for architects have to be living instruments. They need periodic renormalisation as the technology and business landscape shifts, and freezing them in their original form would erode their value over time.

### Limitations and Threats to Validity

Several limitations qualify the contribution. The case is drawn from a single mid-sized company, and while the

matrix's structural design is generalisable, the specific weights and thresholds reflect the operational context of an international consultancy and would not transfer verbatim to other organisations. Reported effects are observations of a complex change initiative in which the matrix was one element among several, and no controlled comparison was possible. Any numerical instrument carries a risk of false precision, because a 0 to 5 rating multiplied by a weight looks objective, while the underlying judgment remains expert and qualitative. The matrix also requires ongoing maintenance as the technical and business landscape shifts, and the cost of this maintenance is itself a non-trivial organisational commitment.

### CONCLUSION

The study demonstrates that assessing the proficiency of software and solution architects requires a competency model that captures the technical, managerial, communicative, and business dimensions of the role within a single evaluative framework. The reviewed professional standards, industry frameworks, and corporate career ladders identify important elements of architectural competence, including career levels, skill categories, and behavioural indicators. Their comparison reveals a gap between broad, universal frameworks and organisation-specific HR instruments, and a need for a matrix that combines skill weighting, criticality tiers, a numeric rating scale, and aggregate thresholds for level confirmation.

The Trinetix case illustrates how such a matrix can be operationalised in a mid-sized international IT company. Six career levels, six competency blocks, and approximately 30 skills form a model in which an architect's progression is expressed through increasing solution scope, more complex client interactions, broader leadership responsibilities, and a stronger connection between architectural decisions and business value. The use of level-specific weights, the classification of skills as Required, Important, or Optional, and the calculation of an aggregate score create an assessment instrument applicable to hiring, promotion, professional development, and project allocation.

The scientific contribution of the article lies in its articulation of a shift from flat descriptions of the architect's role to a multidimensional competency model grounded in actual architecture-practice needs. Its practical contribution lies in showing that a standardised matrix can support a shared language of assessment, reduce dependence on individual interviewer interpretations, facilitate internal mobility, and make architects' development trajectories more transparent within the organisation. The study's limitations stem from its reliance on a single case, the absence of controlled deployment metrics, and the dependence of weights and thresholds on the context of a specific consultancy company. The continued value of such a model depends on periodic revision, as cloud-native architectures, microservices,

enterprise architecture, compliance demands, and generative AI are reshaping the content of architectural proficiency and requiring the assessment instrument to evolve accordingly.

### REFERENCES

1. Assyne, N., Ghanbari, H., & Pulkkinen, M. (2022). The essential competencies of software professionals: A unified competence framework. *Information and Software Technology, 151*, 107020. <https://doi.org/10.1016/j.infsof.2022.107020>
2. Bowers, D., & Sabin, M. (2023). Demonstrating the use of a professional skills framework to support the assessment of dispositions in IT education. *Education and Information Technologies, 29*, 7595–7632. <https://doi.org/10.1007/s10639-023-11933-z>
3. Jabrayilzade, E., Evtikhiev, M., Tüzün, E., & Kovalenko, V. (2022). Bus factor in practice. *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice*, 97–106. <https://doi.org/10.1145/3510457.3513082>
4. Lercher, A., Glock, J., Macho, C., & Pinzger, M. (2024). Microservice API Evolution in Practice: A Study on Strategies and Challenges. *Journal of Systems and Software/the Journal of Systems and Software, 215*, 112110. <https://doi.org/10.1016/j.jss.2024.112110>
5. Martínez-Fernández, S., Bogner, J., Franch, X., Oriol, M., Siebert, J., Trendowicz, A., Vollmer, A. M., & Wagner, S. (2022). Software Engineering for AI-Based Systems: A Survey. *ACM Transactions on Software Engineering and Methodology, 31*(2), 1–59. <https://doi.org/10.1145/3487043>
6. Tavoletti, E., Kazemargi, N., Cerruti, C., Grieco, C., & Appolloni, A. (2021). Business model innovation and digital transformation in global management consulting firms. *European Journal of Innovation Management, 25*(6), 1–41. <https://doi.org/10.1108/ejim-11-2020-0443>
7. Wohlin, C. (2021). Case Study Research in Software Engineering—It is a Case, and it is a Study, but is it a Case Study? *Information and Software Technology, 133*, 106514. <https://doi.org/10.1016/j.infsof.2021.106514>