Computer Life
ISSN: 1819-4818 | Vol. 15, No. 1, 2026

A Dynamic Alignment Mechanism for Curriculum in Higher Vocational Education under Emerging Technology Ecosystems: The Case of HarmonyOS Talent Development

Lian Tong*

School of Information Engineering, Jiangsu Maritime Institute, Nanjing 211170, China
***Corresponding author:** 20178007@jmi.edu.cn

Abstract: The training cycle for technical talent is measured in years, whereas emerging technologies iterate in quarters or months. When the time for a technology to become an essential competence for mainstream industrial positions is shorter than a full program of study, a vocational curriculum characterized by fixed structures remains perpetually in a state of "catching up." Taking the full-scenario distributed operating system ecosystem (HarmonyOS / OpenHarmony) as the technological context and smart ships as the application scenario, this study adopts design-based research to conduct three iterative cycles. It finds that alignment failures are distributed across four levels—training orientation, instructional carriers, collaboration mechanisms, and faculty development—whose common mechanism is the absence of routine demand transmission and system adjustment channels. Accordingly, the study proposes a three-dimensional dynamic alignment framework—"technology trajectory–curriculum architecture–industrial competency map"—together with three design principles: structural decomposability, authentic carriers, and external standards. It further constructs five mechanisms covering curriculum structure, project chains, collaboration platforms, multi-source evaluation, and faculty renewal. Evidence from three iterations shows positive changes in technical competency attainment, engineering practice outcomes, and faculty capability, while exposing unresolved issues such as curriculum updating lagging behind technological evolution; because the pilot class has not yet graduated, long-term effects at the job-performance level await follow-up. The transferable value of this mechanism lies in its operating logic of "demand identification–system adjustment–effect evaluation–recalibration."

Keywords: Emerging technology ecosystem; Dynamic curriculum alignment; HarmonyOS talent development; Smart ship; Competency-based education; Design-based research.

1. Introduction

1.1. Problem Statement

The training cycle for technical talent is measured in years, whereas emerging technologies iterate in quarters or months. When the time for a technology to become an essential competence for mainstream industrial positions is shorter than a full program of study, a curriculum characterized by fixed structures remains perpetually in a state of "catching up." Architectural features of the full-scenario distributed operating system ecosystem (HarmonyOS/OpenHarmony)—such as the distributed soft bus, atomic services, and write-once, deploy-everywhere capability [1]—shift the developer competency baseline as a whole; this is not an addition of course content but a movement of the baseline itself. Constraints tighten further when the technology enters a specific industry: shipboard terminals in smart ships involve multiple heterogeneous protocols, demanding operating environments, and system failures that carry safety implications. External standards such as the *Rules for Smart Ships* [2] both raise the curricular threshold and provide a basis for calibration. This leads to the research question: under the constraints of rigid program structures, lagging faculty knowledge renewal, and limited practical training conditions, how can a technical program in higher vocational education establish a mechanism that keeps its curriculum continuously aligned with the evolving industrial technology ecosystem, without relying on periodic wholesale reconstruction?

1.2. Literature Review and Research Questions

Internationally, several alignment frameworks offer useful references: outcome-based education advocates designing curricula and assessment backward from learning outcomes [3]; constructive alignment requires learning objectives, teaching activities, and assessment tasks to point to the same set of intended outcomes [4]; CDIO uses the complete engineering process as the curricular backbone [5]; and scaffolding theory explains the layered support path for competence development [6]. These frameworks answer "what structure a sound curriculum should have" but not "how curricular structure can remain aligned when the external ecosystem keeps changing"—constructive alignment is a criterion for a design point in time, not a mechanism for continuous operation. Domestic research on industry-education integration has discussed organizational forms of alignment [7], and practical studies in the distributed operating system direction [8][9][10] have verified the feasibility of introducing technical content into courses, but these are mostly limited to single courses and lack system-level alignment mechanism design and empirical testing. Accordingly, three research questions are proposed: RQ1—What aspects do alignment failures manifest in, and what is their common mechanism? RQ2—How can a curriculum and teaching system capable of continuous adjustment be designed? RQ3—What observable changes and unexpected outcomes has this system produced in a real teaching context?

2. Theoretical Framework

Dynamic curriculum alignment refers to the process by which a curriculum, while maintaining structural stability, synchronizes its training objectives, course content, practice carriers, and assessment standards with external demand through institutionalized information channels and adjustment nodes. Its precondition is an **emerging technology ecosystem**—a technology system driven by a leading enterprise or open-source community, in a phase of rapid growth, and without an established stable talent development channel. Its difficulty lies in the **structural constraints of vocational curricula**—program structures and the theory-practice ratio are more rigid than in undergraduate education, and adjusting a course cluster involves linkage among credit allocation, faculty deployment, and practical training conditions, at a cost far higher than adding a single course.

The analytical framework of this study comprises three dimensions and two transmission channels. The **technology trajectory dimension** addresses where the competency baseline is moving: architectural features determine that talent development cannot follow a single-device, single-application training logic, while the pace of version iteration determines that course modules must be decomposable and replaceable. The **curriculum** architecture **dimension** contains four sub-elements—curricular structure, practice carriers, assessment mechanisms, and faculty structure—among which the decomposability of the curricular structure determines the cost of responding to external change. The **industrial competency map dimension** points to the set of competency requirements for real positions, which, beyond general technical competencies, includes industry-specific constraints such as protocol heterogeneity and safety. On the forward channel, changes in the competency map enter curriculum development through school-enterprise collaboration; on the reverse channel, students' performance in real projects and evaluations by enterprise mentors flow back to recalibrate the competency map, while graduates' job performance and employer evaluations constitute the long-term calibration signals for this channel. The two channels form a closed loop of "demand identification–system adjustment–effect evaluation–recalibration," transforming alignment from a one-off design task into an ongoing operating mechanism.

The two channels differ in how quickly they respond and in how strongly they can be relied upon, and the framework must therefore be explicit about the level at which adjustment actually occurs. We distinguish three layers. The **content layer**—the technical points taught within an existing module—can respond within weeks and be updated by a single teacher, but cannot change what the curriculum as a whole prepares students for. The **structure layer**—the boundaries between modules and the sequence of the project chain—responds over a semester and requires coordination among teachers and enterprises, and it is at this layer that the competency map is actually renegotiated. The **institutional layer**—the rules governing how demand is transmitted, who holds assessment weight, and how teacher renewal is recognized—responds over years and largely determines whether the upper two layers keep working once the initial novelty fades. A recurring failure mode observed in practice is to answer a structural demand with a content-layer fix: a new technology is added as a module while the surrounding sequence and assessment remain unchanged, so the adjustment is absorbed by the old structure and the underlying misalignment persists. Making the layer explicit is what allows the framework to decide not only what to change but at what cost, and within what period, the change can realistically be made. Whereas existing frameworks focus on the internal consistency between curriculum and objectives, this study focuses on the consistency between the curriculum and the external ecosystem, and adds the temporal dimension.

3. Research Design

This study adopts design-based research as its overall methodology: RQ2 requires designing a mechanism operable under real constraints, and RQ3 requires evaluating effects in a real context, neither of which can be answered apart from context [11][12]. The study was conducted in the software technology program of a higher vocational institution with distinctive industry characteristics (Jiangsu Maritime Institute), which has long-standing cooperation with the shipping and ship intelligence industry. Smart ships were selected as the application scenario because three constraints—protocol heterogeneity, demanding environments, and safety implications of failure—are strong; if the alignment mechanism can operate under these conditions, the likelihood of transferring it to less constrained domains is higher. The case unit is a 30-student pilot HarmonyOS class (Class 241301) formed in September 2024, together with its supporting curriculum and practical training system. The class is scheduled to graduate in June 2027; as it had not entered the graduation stage during the study period, employment and employer evaluation data were unavailable, and evidence collection focused on the on-campus training process. This constraint shaped the design in a specific way: indicators that can only be observed after graduation were replaced by proximal indicators observable during study, most importantly enterprise mentors' judgments of students' work on real project tasks. These judgments are not a substitute for employer evaluations of graduates, and this study does not treat them as such; they are the earliest available signal from the same industrial source, and the mechanism relies on them precisely because waiting for graduation data would leave the reverse channel silent for three years. The study proceeded through three iterative cycles, each following a "design–implementation–analysis–redesign" loop. Each cycle had an explicit entry point: the first cycle took as its object the mismatch between the existing course sequence and the emerging technology baseline; the second took the newly built project chain and assessment scheme as its object; the third took the stability of the mechanism itself. To keep the adjustment traceable, the research team kept a design log recording each problem identified, the design decision made in response, and the evidence on which the decision rested, so that later cycles could trace why a given arrangement took its current form.

Qualitative data came from semi-structured interviews and classroom observation, targeting multi-party representatives from enterprises, the institution, and students, analyzed thematically. Interviewees included enterprise technical staff and mentors participating in co-building and teaching, program leaders and course teachers, and students in the pilot class; interviews were semi-structured around a fixed topic guide but allowed follow-up questions, and were recorded, transcribed, and coded. Quantitative data included learning process and outcome data, certification and competition

records, and enterprise mentors' evaluations of students' on-campus projects, analyzed with descriptive statistics and pre-post self-comparison. This study set no randomized control group and drew its sample from a pilot class in a single program; quantitative results serve only to describe the direction and magnitude of change, not to infer causality. The study was conducted with the informed consent of participating students, teachers, and enterprise personnel; all participation was voluntary, data were anonymized at the point of collection, and no individual student can be identified in the reported results. Credibility was safeguarded through four measures: triangulation across interview, observation, and outcome data; member checking, in which initial interpretations were returned to participants for confirmation or correction; peer review by colleagues not involved in the design; and an audit trail of the design log and coding records. The researcher was also the designer of the intervention, an inherent tension in design-based research; this was partially mitigated through member checking and peer review, but cannot be fully eliminated, and readers should weigh the findings accordingly.

4. Diagnosing Alignment Failures

4.1. Four Categories of Manifestation

Thematic analysis identified four categories of alignment failure, whose manifestations, mechanism classification, and design responses are shown in Table 1. A common feature of the four is that competency requirements on the technology side have shifted, while training orientation, instructional carriers, collaboration mechanisms, and faculty knowledge structures remain at the previous generation's technology baseline.

Table 1. Manifestations and design responses of alignment failures

Manifestation	Mechanism classification	Design response
Job alignment failure: training orientation misaligned with job competency requirements	Training orientation level: missing competency map	Three-dimensional integrated curriculum structure
Practice carrier failure: experimental content cannot cover architectural features	Instructional carrier level: broken project chain	Three-tier progressive project chain
Collaboration failure: enterprise participation limited to resource input	Collaboration mechanism level: unclear rights and responsibilities	Industry-education collaboration platform
Faculty renewal failure: knowledge structure lags behind the technology stack	Faculty development level: absent renewal channel	Faculty competency renewal mechanism

4.2. Common Mechanism

The four failures form a mutually reinforcing chain through "orientation–carrier–mechanism–faculty": the missing competency map deprives course content of traction and leaves practice carriers without a design basis; unclear rights and responsibilities in the collaboration mechanism prevent enterprise demand from stably entering curriculum development, so the competency map cannot be updated; and the absent faculty renewal channel makes it difficult to translate external demand into teaching content. The critical position is the absence of an alignment channel—curriculum updating is fragmented and event-driven, triggered by a technology hotspot or policy requirement, after which continuous monitoring is lacking until the next round of accumulated pressure. This produces delayed responses, uneven adjustment, and high cost. RQ1 is thereby answered: alignment failures are distributed across the four levels of orientation, carrier, mechanism, and faculty, and their common mechanism lies in the absence of routine demand transmission and system adjustment channels.

5. Designing the Dynamic Alignment System

5.1. Design Principles

Addressing the mechanism above, this study establishes three design principles: **structural decomposability**, where course module boundaries are clear and dependencies explicit, so that a single-point update need not reconstruct the whole; **authentic carriers**, using real or highly simulated industrial projects as the shared carrier for both cultivating and testing competencies, so that teaching content and industrial demand are linked through the same object; and **external standards**, introducing industry technical specifications and enterprise certification standards into the assessment system to form calibration signals independent of internal self-assessment. Together, the three turn alignment from reliance on individual judgment into reliance on institutional operation.

5.2. Curriculum Structure and Project Chain

The curriculum structure is organized along competency progression and comprises three dimensions: courses, projects, and competitions. The course dimension is organized into three stages—foundational, intermediate, and practical—corresponding respectively to development languages and declarative UI frameworks, the distributed soft bus and cross-device transfer, and integrated application development for industry scenarios; boundaries are divided by competency unit so that modules can be replaced without reconstructing an entire stage. The project dimension embeds project tasks at the corresponding level within each module, forming a project chain of "in-class micro-projects–course design projects–comprehensive practical projects." The competition dimension converts competition tasks into teaching resources. The three dimensions point to the same set of competency objectives and accumulate into student portfolios; the course cluster comprises 56 courses totaling 2,570 credit hours. The practical project chain progresses through three tiers: basic verification experiments targeting a single technical point; comprehensive design projects targeting the integration of multiple technical points; and innovative research topics targeting real industrial problems, based on the principle of scaffolding—that as task complexity increases, support is gradually withdrawn [6].

5.3. Collaboration Platform, Assessment, and Faculty Mechanisms

The industry-education collaboration platform is organized at three levels—resource co-construction, technical collaboration, and talent reserve—undertaking respectively three functions: jointly developing certified courses and textbooks, co-building laboratories to undertake real projects, and engaging enterprise mentors in teaching and co-building a professional certification system. Its core lies in transforming enterprises from resource providers into co-

educating subjects. The assessment system covers four dimensions—knowledge, skills, innovation, and professional conduct—emphasizing multi-source participation (enterprise mentors hold substantive assessment weight in comprehensive design projects and innovative research topics), process dynamism (incorporating three types of data—learning process, outcomes, and perceptions—which also constitute inputs to the reverse channel), and contextualized content (introducing industrial standards such as code conventions, protocol adaptation correctness, and system stability). Faculty development comprises three arrangements: mixed school-enterprise staffing, selecting teachers for advanced enterprise technical training and certification, and steering teachers toward cross-domain competency mentors. If teachers lack the ability to translate new demand into teaching content, the channel's output cannot be realized.

Two features of the design deserve emphasis because they bear on whether it can keep operating. First, the modules are governed by version management: each module carries a version number, a dated record of what changed, and the evidence that triggered the change, so that an update is a traceable operation rather than an ad hoc revision, and so that later cohorts inherit a documented state rather than an oral tradition. Second, the review arrangement is joint: course boundary and content changes are reviewed by teachers and enterprise technical staff together on a fixed cycle, rather than being initiated by the institution and then submitted to enterprises for endorsement. The distinction matters because the first arrangement keeps demand transmission routine and low-cost, whereas the second reduces it to an occasional approval event—and it was the absence of routine transmission that Chapter 4 identified as the common mechanism of failure. The two features are deliberately modest in scope: neither requires new institutional authority, and both can be implemented with existing teaching and collaboration arrangements.

6. Implementation Process and Effect Evidence

6.1. Three Iterative Cycles

The system was advanced in three cycles using the pilot class as its carrier, over an overall period from September 2024 to September 2026, each cycle about one semester. The first cycle (September 2024–June 2025) focused on restructuring the curriculum and building practice carriers, finding a gap in competency requirements between the three tiers of projects—students showed a competency discontinuity when moving from verification experiments to comprehensive projects—and the second cycle accordingly added transitional tasks. The decisive evidence was that in the first rounds of comprehensive design projects, students could complete each technical point in isolation but could not decide which technical points a given requirement implied; the discontinuity was therefore one of design judgment rather than of operation, and the transitional tasks were built around requirement decomposition rather than around additional hands-on practice.

The second cycle (September 2025–June 2026) focused on embedding the assessment mechanism and optimizing the project chain, finding two unexpected issues: enterprise project resource supply fluctuated with business rhythms, and students' distributed systems thinking took longer to develop than expected. Neither had been anticipated in the design: the first is an organizational rather than technical problem, and the second inverted an assumption in the original sequence, which had placed distributed content relatively late on the reasoning that students should first be secure in single-device development. Observation suggested the opposite—students who encountered cross-device scenarios early formed a more accurate mental model of the architecture, whereas those who met it only after becoming fluent in single-device development tended to treat cross-device behavior as an exceptional case.

The third cycle (from September 2026) accordingly added supplementary sources of in-house projects and adjusted the entry point of distributed competency content, while establishing a version management system for course modules and a joint school-enterprise curriculum review arrangement. Notably, the third cycle's changes were not triggered by new external demand but by the mechanism's own operating record—the failures it had surfaced in the previous two cycles. This is the clearest available evidence that the loop described in Chapter 2 is functioning: the system is generating inputs to its own redesign rather than merely responding to external shocks.

6.2. Effect Evidence and Reliability Qualifications

The system's operating effects were observed across five aspects—technical competency attainment, innovation and competition outcomes, enterprise recognition, faculty capability improvement, and teaching resource development—with indicators shown in Tables 2 and 3. The enterprise recognition dimension substitutes enterprise mentors' ratings of comprehensive projects for graduate employment data: the pilot class graduates in June 2027, so job performance evaluation requires post-graduation follow-up collection, and this study lists it as a subsequent validation stage of the mechanism rather than current evidence.

Table 2. Effect indicators of the talent development system

Dimension	Indicator	Value	Source
Technical competency	Average pass rate (by course enrollment, n=150)	98.7% (148/150)	Academic affairs system
	Project completion rate / number of works	100% / 90	Teaching records
Innovation and competition	Awards at or above provincial level	National: 2 first, 3 second, 2 third, 1 excellence; provincial: 4 first, 9 second, 22 third	Award certificates
	Enterprise technical certification passes	30 students, 100%	Certification records
	Enterprise mentors' rating of projects	Mean 97 (out of 100, n=3)	Mentor evaluation forms

Four qualifications apply when interpreting this evidence: no control group was set, so pre-post changes cannot exclude the joint effects of students' own maturation, teachers' accumulated experience, and other curriculum reforms; some

indicators come from self-reported data, subject to social desirability bias; the sample comes from a single pilot class at a single institution; and the intervention targets a rapidly evolving technology whose architecture kept changing during the study, so the baseline for pre-post comparison is not entirely fixed. In addition, the case class has not yet graduated, so its job adaptability and employer evaluations cannot be examined in this paper; the long-term calibration effect of the reverse channel awaits follow-up. This section's evidence should therefore be understood as descriptive evidence of the direction of change rather than causal proof of effects.

Table 3. Capacity- and resource-building indicators

Dimension	Indicator	Value	Source
Faculty capability	Person-times in enterprise training and certification	12	Training and certification records
	Related projects applied for and teaching papers published	3 projects, 4 papers	Research records
Teaching resources	Online open courses and new-format textbooks	5 online courses, 2 textbooks	Resource repository

7. Discussion and Conclusion

This study yields three findings. First, **mechanisms are more transferable than content**: the course content underwent multiple rounds of adjustment, whereas the operating mechanism supporting that adjustment remained essentially stable across the three iterations, suggesting that the core asset an institution can accumulate is not any version of course content but the capability to organize content updating. Second, **external standards are the key constraint for maintaining alignment**: internally devised assessment standards tend to drift downward, taking students' actual level rather than job requirements as the reference; after introducing enterprise certification standards and industry technical specifications, the alignment between teaching content and industrial requirements improved markedly, indicating that when the alignment target is moving, the anchor must come from outside the system [4]; this study uses enterprise mentors' evaluation of on-campus projects as proximal evidence of external calibration, and whether it agrees with post-graduation job performance requires follow-up verification. Third, **dynamic alignment incurs costs**: modular design, project resource maintenance, enterprise mentor coordination, and assessment data collection all require additional investment, which in this study was partly shared by partner enterprises and partly absorbed through reorganizing existing teaching activities.

Compared with existing research, this study's positioning lies in three respects: it extends rather than replaces the constructive alignment principle, which concerns internal curricular consistency, whereas this study concerns consistency between the curriculum and the external ecosystem; it adds a temporal dimension to outcome-based education and the CDIO framework [3][5]; and it advances beyond published teaching research in the distributed operating system direction, where existing studies verified the feasibility of introducing technical content into a single course, whereas this study answers, at the system level, "how to maintain alignment after entering." The mechanism is transferable to domains meeting three conditions: an accessible industry collaboration channel, a decomposable curriculum structure, and externalable standards; its boundary is that when a technology ecosystem has not yet formed a stable job competency map, institutions should observe and prepare rather than fully restructure.

In summary, three main conclusions follow: the common mechanism of alignment failure lies in the absence of routine demand transmission and system adjustment channels, so the focus of reform should shift from "which course to change" to "what kind of adjustment channel to establish"; the three-dimensional dynamic alignment framework and the three design principles constitute a temporal-dimension supplement to existing curriculum design frameworks; and evidence from three iterations indicates that the system showed positive changes in technical competency attainment, engineering practice outcomes, and faculty capability, while exposing unresolved issues such as curriculum updating still lagging behind technological evolution. Limitations are: as a single-case study, external validity awaits verification across multiple institutions; without a control group, quantitative evidence can only describe change and cannot support causal inference; the case class has not yet graduated, so the long-term calibration effect of the reverse channel (graduates' job performance and employer evaluations) has not been tested and awaits post-graduation follow-up; and the researcher is also the designer, an inherent tension partially mitigated but not fully eliminated through member checking and peer review.

Acknowledgment

This work was supported by the funding of the University-Level Research Project of Jiangsu Maritime Institute, "Theory and Practice of the Training Model for Software Technology (HarmonyOS Direction) Talent Adapting to New Quality Productive Forces" (No. 2025JYyb03), the University-Level Teaching Reform Project, "Construction and Practice of a Composite HarmonyOS Talent Development System under the Smart Ship Ecosystem", and the School-Level Teaching Reform Project of the School of Information Engineering, "Research and Practice of Project-Based Teaching Reform in the Course Front-End Framework Application Development".

References

[1] Huawei Technologies Co., Ltd. (2025). HarmonyOS ecosystem solution white paper (2025 edition) [White paper].

[2] China Classification Society. (2020). Rules for smart ships [Standard]. China Classification Society.

[3] Spady, W. G. (1994). Outcome-based education: Critical issues and answers. American Association of School Administrators.

[4] Biggs, J. (1996). Enhancing teaching through constructive alignment. Higher Education, 32(3), 347-364. https://doi.org/10.1007/BF00138871

[5] Crawley, E. F., Malmqvist, J., Östlund, S., Brodeur, D. R., & Edström, K. (2014). Rethinking engineering education: The CDIO approach (2nd ed.). Springer. https://doi.org/10.1007/978-3-319-05561-9

[6] Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. Journal of Child Psychology and Psychiatry, 17(2), 89-100. https://doi.org/10.1111/j.1469-7610.1976.tb00381.x

[7] Zhang, P., & Nan, X. G. (2024). Coexistence and coupling: High-quality development of vocational education from the perspective of new quality productive forces. Journal of Higher Education Management, (18), 1-10.

[8] Cheng, B., & Zhang, B. Y. (2024). Exploration of teaching reform in embedded systems courses based on OpenHarmony. Open Access Library Journal, 11, Article e12692. https://doi.org/10.4236/oalib.1112692

[9] Tang, J. Q. (2025). Exploring interdisciplinary project-based teaching reform of "AI meets HarmonyOS" in higher-vocational IoT programs. Open Journal of Social Sciences, 13, 336-348. https://doi.org/10.4236/jss.2025.136023

[10] Zhu, Y., Jing, R., & Tang, J. (2026). The curriculum reform and practical exploration of HarmonyOS application development under the background of emerging engineering education. ERD, 8(7), 50-56.

[11] Design-Based Research Collective. (2003). Design-based research: An emerging paradigm for educational inquiry. Educational Researcher, 32(1), 5-8. https://doi.org/10.3102/0013189X032001005

[12] Barab, S., & Squire, K. (2004). Design-based research: Putting a stake in the ground. Journal of the Learning Sciences, 13(1), 1-14. https://doi.org/10.1207/s15327809jls1301_1