The 90 to 120 Day Term Sheet

How UAE Developers Can Compress Time-to-Capital in 2026

Authored by CK Adya

Independent Researcher, London Business School

May 2026

Abstract

Speed of access to capital is a competitive variable that United Arab Emirates (UAE) real estate developers routinely underestimate. While much attention is paid to the price and structure of capital, far less is paid to the time it takes to secure it, yet that time carries a real and quantifiable cost: carrying costs accrue, acquisition opportunities slip to faster competitors, and the eventual return is eroded by every week of delay. This paper argues that the time from launching a capital raise to a signed term sheet, which for an unprepared developer can stretch to many months, can be compressed to ninety to one hundred and twenty days through deliberate preparation, parallel process design and disciplined execution, and that doing so is a genuine source of competitive advantage. Using an indicative dataset calibrated to 2026 GCC conditions, the study sets out the anatomy of the capital-raising timeline, develops a data-room and readiness framework that is the single largest determinant of speed, contrasts parallel and sequential process designs, examines the compression of due diligence including the role of artificial intelligence, and quantifies the cost of delay. The analysis finds that data-room readiness and process parallelism dominate the time-to-capital, that the cost of delay compounds and the probability of deal slippage rises sharply beyond the plan, and that a developer that institutionalises speed as a capability rather than improvising each raise can systematically reach a term sheet in the ninety to one hundred and twenty day window. Three indicative case studies, a sensitivity analysis, an international comparison, a technology architecture for an accelerated raise, and an implementation roadmap support the framework.

Keywords: Capital raising, data room, deal execution, due diligence, real estate development, term sheet, time-to-capital, UAE

1. Introduction

When developers discuss capital, they discuss its price and its structure, the margin on the senior debt, the coupon on the mezzanine, the share of equity conceded to a partner. They discuss its speed far less often, and yet the speed at which capital can be secured is frequently the difference between capturing an opportunity and losing it. A developer that can present a signed term sheet within ninety days can move on an acquisition that a developer requiring six months cannot, and in a competitive market that speed advantage is worth more than a marginal saving on the cost of the capital itself.

This paper treats time-to-capital as a variable to be managed deliberately rather than a residual to be endured. It asks how long it should take a UAE developer to move from launching a capital raise to a signed term sheet, what determines that time, and how it can be compressed. The central claim is that the ninety to one hundred and twenty day term sheet is achievable for a prepared developer, that the gap between that benchmark and the many months an unprepared

developer takes is largely self-inflicted, and that closing it is one of the highest-return process improvements a developer can make.

The argument rests on a simple observation: most of the delay in a capital raise is not caused by the capital providers but by the developer. Incomplete data rooms, models that do not reconcile, missing approvals, sequential processes that could have run in parallel, and slow internal decision-making account for the majority of the elapsed time, and all of these are within the developer control. The capital provider response time, the part the developer cannot control, is a minority of the total. The opportunity, therefore, is to attack the self-inflicted delay, and the framework in this paper is a structured approach to doing so.

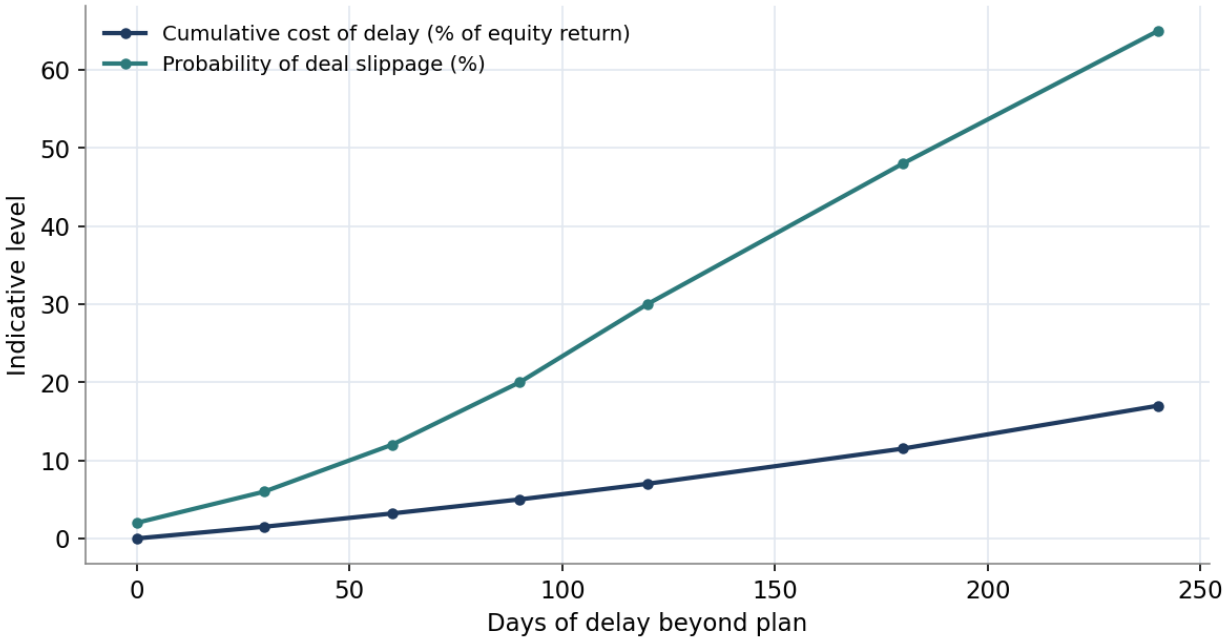
The figures used throughout are indicative, calibrated to observable GCC conditions in early 2026 but not drawn from any specific transaction. The paper proceeds from why time-to-capital matters (Section 2), through the conceptual foundations of the fundraising process (Section 3), the anatomy of the timeline (Section 4), the ninety to one hundred and twenty day benchmark (Section 5), the readiness framework (Section 6), parallel process design (Section 7), the term sheet (Section 8), diligence compression (Section 9), the causes of delay (Section 10), UAE-specific considerations (Section 11), the capital-provider perspective (Section 12), three case studies (Section 13), sensitivity analysis (Section 14), an international comparison (Section 15), the role of technology (Section 16), the cost of delay (Section 17), common errors (Section 18), an implementation roadmap (Section 19), a strategic perspective (Section 20), a conclusion (Section 21) and limitations (Section 22).

2. Why Time-to-Capital Matters

The cost of a slow capital raise is real even though it rarely appears as a line item. The first component is the carrying cost: every week that a developer holds land, or delays a construction start, while it raises capital is a week of financing cost, holding cost and overhead borne against an asset that is not yet generating a return. For a developer carrying expensive land-stage or bridge capital, the carrying cost of a slow raise can be substantial, and it accrues regardless of whether the raise ultimately succeeds.

The second and often larger component is opportunity cost in the form of deal slippage. Real estate opportunities, whether an attractive land parcel, a distressed asset or a time-limited acquisition, are frequently competitive and time-limited, and a developer that cannot move quickly loses them to a faster rival. The lost opportunity does not appear in any account, but it is a genuine cost: the developer forgoes the return it would have earned on the deal it could not move on in time. Over a series of opportunities, a chronic inability to move quickly compounds into a meaningful drag on growth.

Figure 1. The Compounding Cost of Delay and the Rising Probability of Slippage



Cost of delay accumulates and slippage probability rises with days beyond plan. Not a forecast.

Figure 1 illustrates both components. The cumulative cost of delay rises steadily with each day beyond the plan, as carrying costs accrue, while the probability that the deal slips away entirely rises sharply, because counterparties lose patience, markets move and competitors intervene. The two curves together show why time-to-capital is not a soft consideration but a hard determinant of outcomes: delay both erodes the return on the deals that close and raises the chance that deals do not close at all. A developer that ignores time-to-capital is, in effect, accepting an invisible but substantial tax on every transaction.

There is also a less tangible cost in the developer reputation and relationships. A developer known to capital providers as slow, disorganised or unreliable in execution finds that providers prioritise faster, cleaner counterparties, that terms drift against it, and that opportunities are brought to it last. A developer known to be fast, prepared and reliable finds the opposite: providers compete for its business, terms improve, and opportunities are brought to it first. Speed, in other words, compounds into a reputational asset that improves the developer access to capital over time, and slowness compounds into a reputational liability.

A useful way to internalise the cost of slowness is to express it as an implicit interest rate on time. If a developer can show that each additional month of delay costs it a given percentage of its eventual return, then time itself acquires a price, and decisions that trade money for time, paying for vendor diligence, engaging an adviser, investing in a data-room platform, can be evaluated against that price. Many developers intuitively treat time as free and money as scarce, and so under-invest in the things that buy speed; the discipline of pricing time corrects this, revealing that modest expenditures which compress the timeline are frequently among the highest-return decisions in the whole transaction.

The cost of delay is also asymmetric in an important way. A raise that completes a month early rarely produces a windfall, because the opportunity it funds is usually available on the planned timeline in any case, but a raise that completes a month late can forfeit the opportunity entirely. Speed therefore protects against downside more than it generates upside, which means its value is

best understood as insurance against losing opportunities rather than as a source of extra return on the opportunities that close. This framing matters because insurance is easy to under-value until the loss it protects against actually occurs, and developers that have lost a deal to slowness value speed far more highly than those who have not yet experienced the loss.

3. Conceptual Foundations: The Capital Raise as a Pipeline

The capital raise is usefully understood as a pipeline, a sequence of stages through which a transaction passes from launch to funding, with a conversion rate and a cycle time at each stage. This framing, borrowed from the analysis of sales and operational processes, allows the developer to think about time-to-capital systematically rather than anecdotally, identifying which stages consume the most time and which can be compressed. The total time-to-capital is the sum of the cycle times of the stages, adjusted for any parallelism, and the task of compression is to reduce the cycle time of each stage and to overlap stages where possible.

A pipeline framing also draws attention to the conversion rate at each stage, the proportion of opportunities that advance from one stage to the next. A raise that reaches many capital providers but converts few into term sheets is inefficient, consuming time without progressing, and the remedy may be better targeting rather than faster execution. Conversely, a raise that converts well but reaches too few providers may simply need more outreach. Understanding both the cycle time and the conversion rate at each stage allows the developer to diagnose where its raises are slow and why, and to direct its improvement effort accordingly.

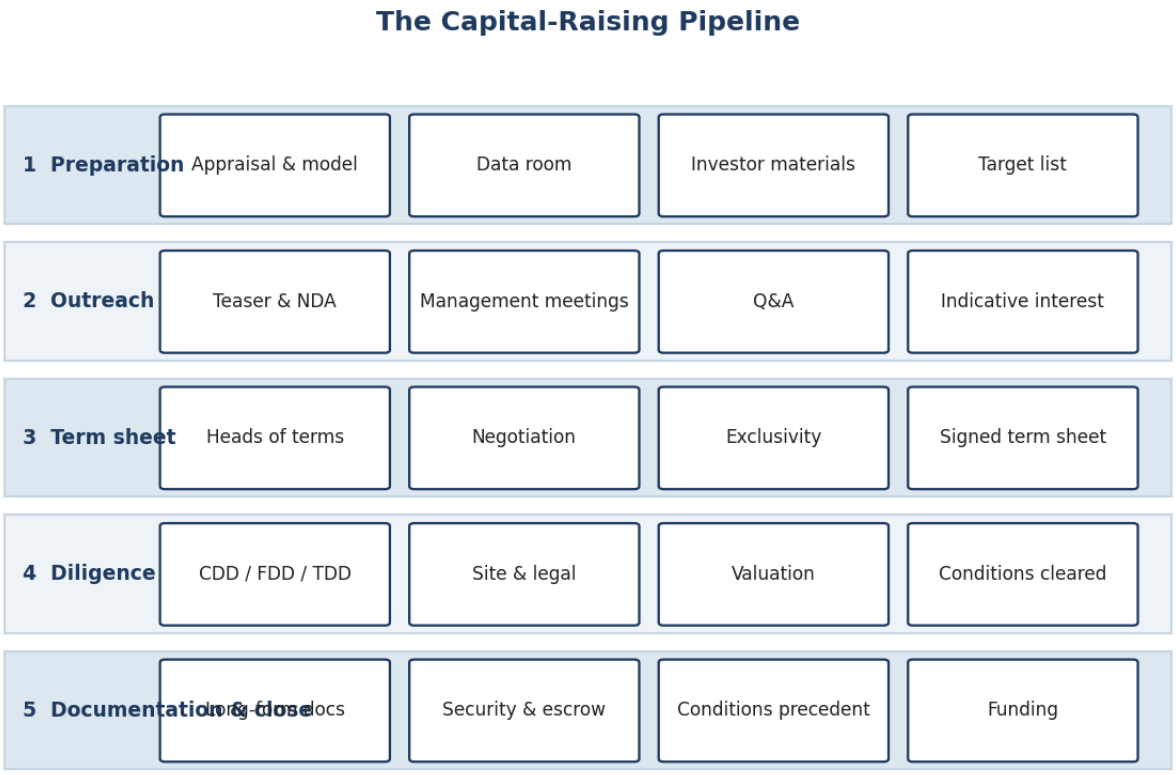
The deeper conceptual point is that speed has an option value. A developer that can raise capital quickly holds, in effect, an option to act on opportunities as they arise, and that option has value precisely because opportunities are uncertain and time-limited. The developer that must spend months raising capital before it can act has, in effect, no such option, and must either commit capital speculatively in advance or miss opportunities that require a fast response. The capability to raise capital quickly is therefore not merely an efficiency but a strategic asset, and investing in it is investing in the option to act, which is worth more the more volatile and competitive the market.

The pipeline framing yields a practical diagnostic. A developer that records the cycle time and the conversion rate of each of its raises over time builds a dataset that reveals where its process is slow and where it leaks. Perhaps its outreach converts poorly because its targeting is weak; perhaps its term-sheet stage drags because it comes to negotiations without a clear position; perhaps its diligence stalls because its data rooms are incomplete. Without this measurement, a developer is left to improve its process by intuition; with it, the developer can direct its effort precisely at the stage that most constrains its speed. The measurement itself is inexpensive, requiring only the discipline to record the dates and outcomes of each stage, and it converts process improvement from guesswork into management.

4. Anatomy of the Capital-Raising Timeline

The capital-raising timeline can be divided into five stages, illustrated in Figure 2: preparation, outreach, term sheet, diligence and documentation. Each has a characteristic duration and a characteristic set of activities, and understanding them is the prerequisite to compressing the whole.

Figure 2. The Five Stages of the Capital-Raising Pipeline



Indicative mapping of stages and activities. Not transaction-specific.

The preparation stage produces the materials a capital provider needs to engage: the development appraisal and financial model, the data room, the investor presentation and the target list of providers. This stage is entirely within the developer control, and it is where the largest time savings are available, because a developer that prepares thoroughly before launching can move through the subsequent stages far faster than one that prepares as it goes. The outreach stage takes the prepared materials to the target providers, holds management meetings, answers questions, and elicits indicative interest. Its duration depends on the quality of the materials and the targeting, since well-targeted outreach to the right providers with complete materials converts quickly to interest.

The term sheet stage negotiates and signs the heads of terms that set out the principal commercial terms, and it is the milestone this paper takes as its benchmark, because a signed term sheet represents genuine commitment subject to diligence. The diligence stage sees the provider conduct its commercial, financial, technical and legal due diligence and clear its conditions, and it is frequently the longest stage, though as Section 9 shows it can be compressed. The documentation stage negotiates the long-form legal documents, puts the security and escrow in place, satisfies the conditions precedent and funds the transaction. The total elapsed time is the sum of these stages, less any overlap, and the benchmark of ninety to one hundred and twenty days to a term sheet covers the first three stages, with diligence and documentation following.

It is worth noting that the stages are not equally compressible. The preparation stage can be largely eliminated from the critical path by completing it before launch; the outreach and term sheet stages can be accelerated by good targeting and materials; the diligence stage can be compressed by readiness and technology; and the documentation stage, which depends heavily on legal process and counterparty behaviour, is the least compressible. A realistic compression

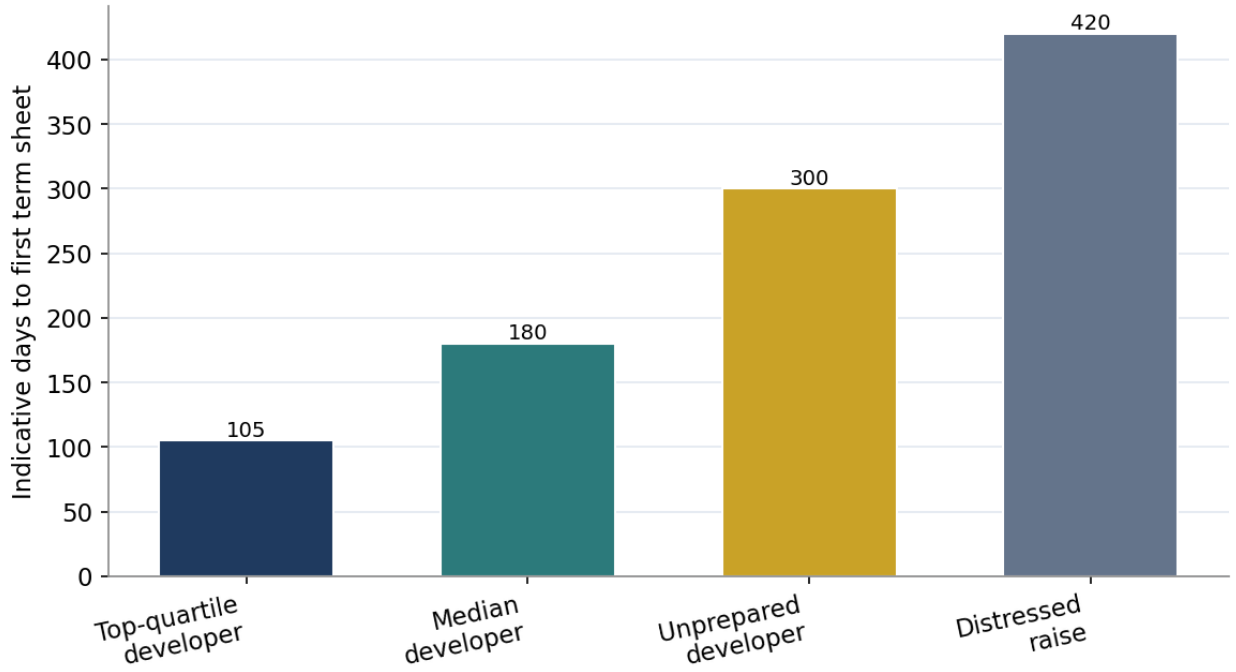
strategy therefore front-loads the effort into preparation and process design, where the returns are highest, rather than attempting to force the least compressible stages.

The five stages also differ in who controls them, which matters for where the developer should focus. Preparation is entirely within the developer control; outreach is largely within its control, subject to provider availability; the term-sheet stage is shared, depending on both the developer position and the provider appetite; diligence is largely provider-controlled, though enabled by developer readiness; and documentation is shared with the lawyers on both sides. The developer therefore has the most leverage over the early stages, which is fortunate, because the early stages are also where the largest time savings are available. The strategic implication is to invest heavily in the stages the developer controls and can compress, rather than fretting over the later stages where its leverage is limited.

5. The 90 to 120 Day Benchmark

The central benchmark of this paper is that a prepared developer should reach a signed term sheet within ninety to one hundred and twenty days of launching a raise. Figure 3 places this benchmark in context, comparing the time to first term sheet across developers of different levels of preparation.

Figure 3. Indicative Days to First Term Sheet by Developer Preparedness



The figure shows a wide dispersion. A top-quartile, well-prepared developer reaches a term sheet in around one hundred days; a median developer takes closer to one hundred and eighty; an unprepared developer takes three hundred or more; and a developer raising under distress, with a weak position and incomplete materials, can take well over a year, by which time the opportunity that prompted the raise has often passed. The gap between the top quartile and the median is the prize: it represents roughly eighty days of elapsed time, and the carrying cost and opportunity cost of those eighty days, that a developer can capture purely by improving its preparation and process.

The benchmark is achievable because the binding constraint on speed is usually the developer own readiness, not the capital provider response time. A provider presented with a complete, well-organised data room, a model that reconciles, clear title and approvals, and a credible sponsor can move to a term sheet quickly, because it has everything it needs to form a view. A provider presented with gaps, inconsistencies and missing documents must wait while the developer fills them, and each gap adds days or weeks. The ninety to one hundred and twenty day benchmark is therefore a statement about developer readiness as much as about market conditions: it is the time a raise takes when the developer has removed the self-inflicted delays.

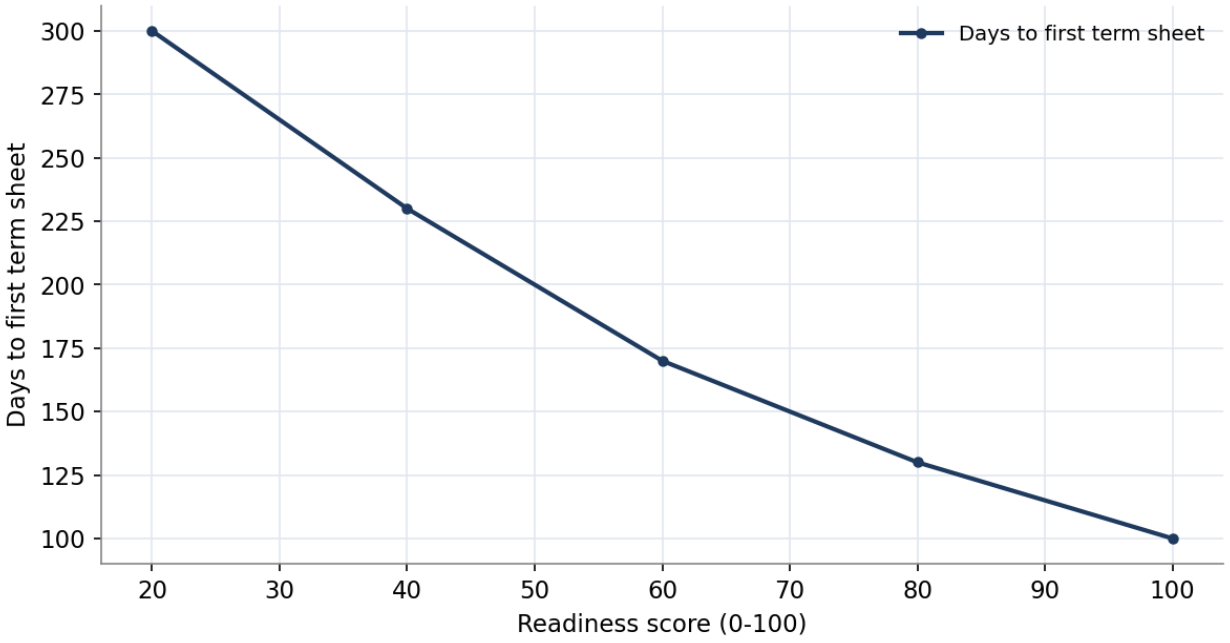
The benchmark should be understood as a target for the term sheet, not for funding. The diligence and documentation that follow the term sheet add further time, typically a further sixty to ninety days, so that a raise reaching a term sheet in one hundred days might fund in around one hundred and eighty. But the term sheet is the critical milestone, because it represents committed interest and allows the developer to act on the opportunity with confidence, and it is the milestone most within the developer power to accelerate. The remainder of this paper focuses on compressing the time to the term sheet, while noting where the subsequent stages can also be accelerated.

It is worth distinguishing the benchmark from a stretch target. Ninety to one hundred and twenty days is not the fastest conceivable raise but the consistently achievable one for a prepared developer in normal conditions; exceptional raises, with a single eager provider and a simple structure, can close faster still, while complex portfolio or cross-border raises may reasonably take longer. The value of the benchmark is as a standard against which a developer can measure itself and diagnose its shortfalls, not as a rigid deadline. A developer consistently taking twice the benchmark should treat that gap as a signal of a process problem to be investigated, while a developer near the benchmark can focus its energy on the terms rather than the timeline.

6. The Data-Room and Readiness Framework

The single largest determinant of time-to-capital is the developer readiness at the moment it launches the raise, and readiness is embodied above all in the data room. Figure 4 shows the strong relationship between a developer readiness, scored against a structured checklist, and the days it takes to reach a first term sheet.

Figure 4. Days to First Term Sheet Against Readiness Score



Higher readiness sharply reduces time to term sheet. Not a forecast.

The relationship is steep and consistent: a developer scoring highly on readiness reaches a term sheet in around one hundred days, while a poorly prepared developer takes three times as long. The readiness score aggregates the completeness and quality of the materials a provider needs: the appraisal and model, the title and registration documents, the approvals and no-objection certificates, the construction and cost information, the sales and market evidence, and the corporate and governance documents. Each gap in these materials is a question the provider must raise and the developer must answer, and each such exchange adds days. A complete data room pre-empts the questions and removes the delay.

A readiness framework therefore consists of a structured checklist of the materials a provider will require, assembled and quality-checked before the raise is launched. The discipline is to treat the data room as a product to be completed before going to market, rather than as something assembled reactively in response to provider questions. A developer that completes the data room first launches from a position of readiness and moves quickly; a developer that launches with an incomplete data room spends the subsequent weeks assembling materials under pressure, which is both slower and more error-prone. Appendix A sets out an indicative readiness checklist.

Crucially, readiness is not merely about completeness but about consistency and credibility. A model that does not reconcile to the appraisal, a title document that does not match the description, or a cost plan that is inconsistent with the programme will raise questions and erode the provider confidence, slowing the raise even if every document is present. The readiness framework therefore includes a quality-control pass that checks the materials for internal consistency and credibility before launch, because a provider that finds one inconsistency begins to doubt the rest, and the resulting scrutiny slows everything. Readiness is the combination of completeness, consistency and credibility, and all three must be achieved before launch.

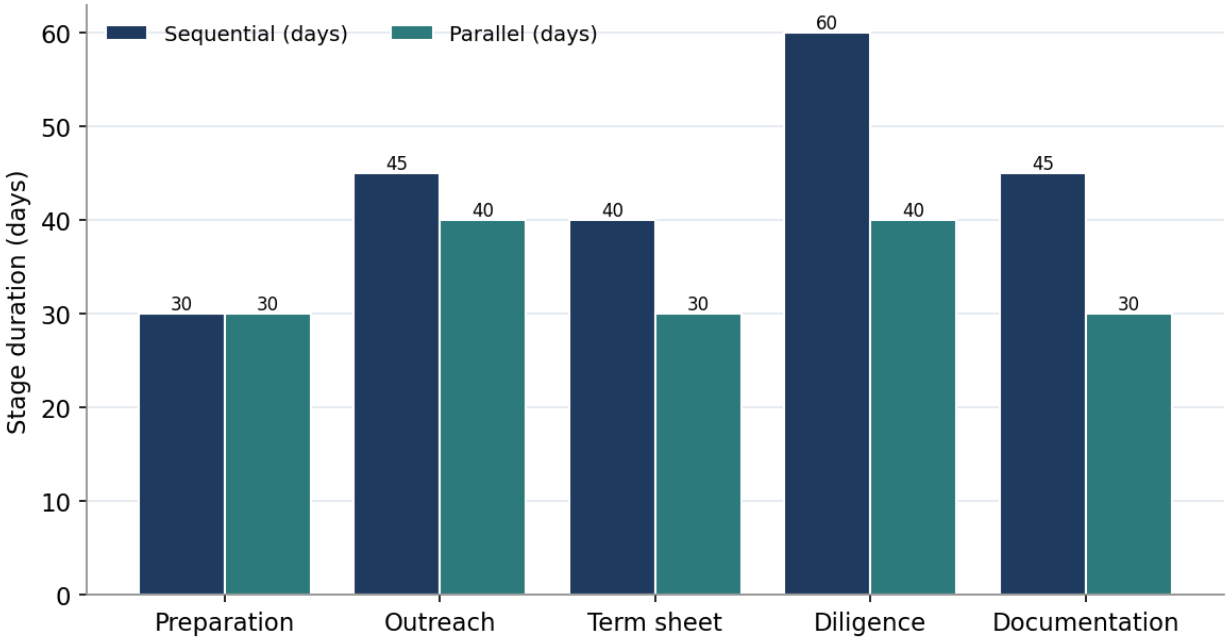
A subtle benefit of high readiness is its effect on provider confidence and therefore on terms as well as speed. A provider presented with a complete, consistent, credible data room infers that the developer is competent and well-run, and extends that inference to its assessment of the project

and the sponsor, which supports both a faster decision and better terms. A provider presented with gaps and inconsistencies infers the opposite, and prices the perceived execution risk into both its timeline and its terms. Readiness, in other words, is not merely an operational efficiency but a signal of quality, and the developer that invests in it is rewarded twice, in the speed of the raise and in the terms it achieves.

7. Parallel versus Sequential Process Design

After readiness, the second largest lever on time-to-capital is the design of the process, and specifically the choice between running the stages and the capital providers sequentially or in parallel. A sequential process, approaching one provider at a time and completing each stage before beginning the next, is slow and cedes negotiating leverage; a parallel process, approaching multiple providers simultaneously and overlapping stages where possible, is faster and creates competitive tension. Figure 5 compares the two.

Figure 5. Stage Durations Under Sequential and Parallel Process Designs



Parallel processes overlap stages and shorten the total. Not a forecast.

The figure shows the parallel process shortening every stage, but the largest savings come from overlapping the stages rather than running them strictly in sequence. Outreach to multiple providers can begin while preparation is being finalised; term-sheet negotiations with several interested providers can proceed simultaneously; and preliminary diligence can begin before the term sheet is signed. The parallel process also creates competitive tension: providers aware that they are competing move faster and offer better terms, which compresses the term-sheet stage and improves the outcome. The sequential process, by contrast, gives each provider time and leverage, and forfeits the discipline that competition imposes.

Running a parallel process requires more capacity and organisation from the developer, because it must manage multiple providers, multiple sets of questions and multiple negotiations simultaneously. This is precisely why readiness matters: a developer with a complete data room can service multiple providers in parallel from the same materials, while a developer assembling materials reactively cannot, and is forced into a sequential process by its own lack of preparation.

Readiness and parallelism are therefore complementary: readiness enables parallelism, and parallelism is what converts readiness into speed. A developer that achieves both can run a fast, competitive process; a developer that achieves neither is condemned to a slow, sequential one.

There is a judgement to be made about how many providers to run in parallel. Too few forfeits competitive tension and risks the process stalling if a provider drops out; too many dilutes the developer attention, signals desperation, and risks the market learning that the developer is shopping the deal widely. The right number depends on the size and attractiveness of the raise, but a focused parallel process of a handful of well-targeted providers usually optimises the balance between competition and focus. The targeting matters as much as the number: a parallel process of the right providers converts quickly, while a parallel process of poorly matched providers wastes effort on parties that will not transact.

8. The Term Sheet: Anatomy and Negotiation

The term sheet is the milestone the developer is racing toward, and understanding its anatomy is essential to negotiating it quickly. A term sheet sets out the principal commercial terms of the proposed financing: the amount, the structure, the pricing, the security, the key conditions, the exclusivity and the indicative timetable. It is not the final binding agreement, which comes with the long-form documentation, but it represents genuine commitment subject to diligence, and it allows the developer to act on its opportunity with confidence. Table 1 sets out the principal components.

Table 1. Anatomy of a Financing Term Sheet

Component	What it covers	Why it matters for speed
Amount and structure	Quantum, tranches, ranking	Defines the deal; must match the appraisal
Pricing	Margin, fees, coupon	Often the most negotiated; prepare a range
Security	Charges, pledges, escrow	Clarify early to avoid diligence surprises
Conditions	Conditions precedent, approvals	Identify and pre-clear where possible
Exclusivity	Period and scope	Balances commitment against competition
Timetable	Indicative path to close	Sets expectations and discipline

A prepared developer can negotiate these quickly from a clear prior position. Not transaction-specific.

The speed of the term-sheet negotiation depends heavily on how clearly the developer has thought through its own position in advance. A developer that knows the structure it wants, the pricing range it will accept, the security it is willing to grant and the conditions it can satisfy can negotiate quickly and decisively, because it is not working out its position as it goes. A developer that comes to the negotiation without a clear prior position must deliberate over each term, which slows the negotiation and signals uncertainty to the provider. Preparation, again, is the key: the developer that has done the thinking in advance negotiates fast, while the developer that improvises negotiates slowly.

A common cause of delay at the term-sheet stage is the valuation or pricing gap, where the developer and the provider hold materially different views of the value or the appropriate pricing. This gap is best surfaced and addressed early, during outreach, rather than allowed to emerge late in the term-sheet negotiation, because a gap discovered late wastes the time invested in reaching that point. A developer that presents credible, well-evidenced valuation and pricing expectations during outreach filters out providers whose views are too far apart and focuses on those with whom a deal is possible, compressing the term-sheet stage by avoiding negotiations that were never going to converge.

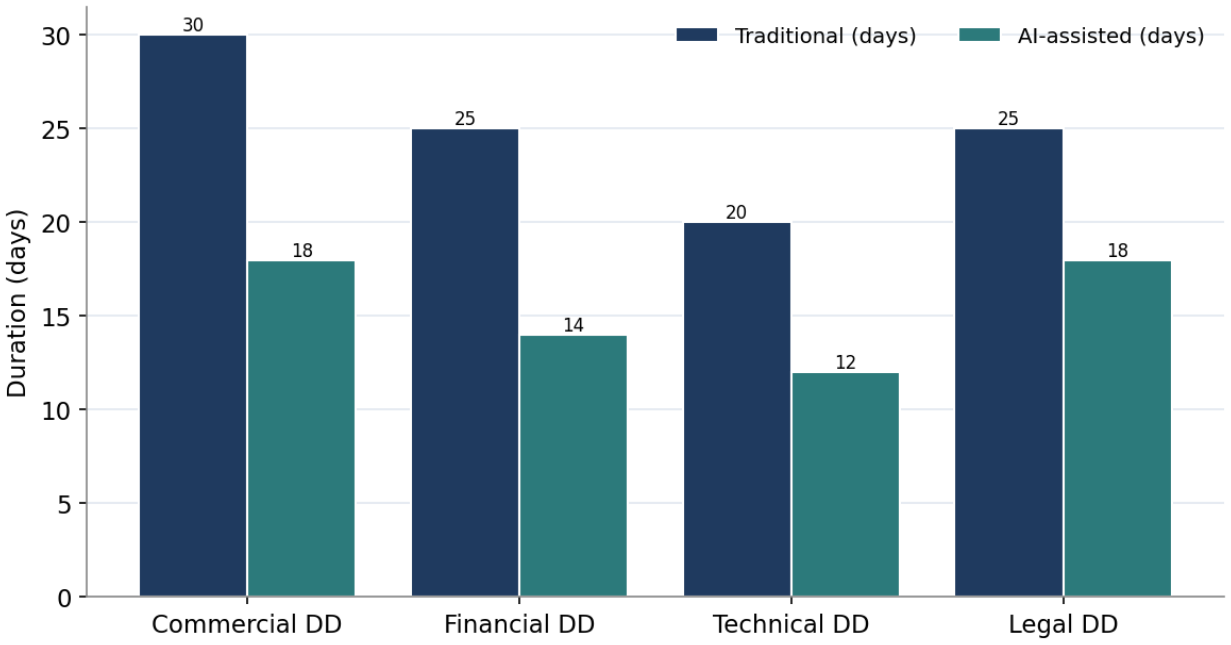
Exclusivity deserves careful handling because it trades speed against competition. A provider will often seek exclusivity before committing to diligence, to protect its investment of time and cost; granting it removes the competitive tension that disciplines terms. The developer should grant exclusivity only when the term sheet is sufficiently firm and favourable to justify it, and should keep the exclusivity period short and conditional on the provider meeting the timetable, so that a provider that delays loses its exclusivity and the developer can revert to the competing providers. Managed well, exclusivity accelerates the committed provider without forfeiting the developer leverage; managed badly, it stalls the process by tying the developer to a single, now-unpressured provider.

The judgement about parallelism extends to how the developer sequences the providers it most and least wants. A common and effective approach is to run the most-preferred providers slightly ahead, so that a term sheet from a preferred provider arrives first and can be used to discipline the others, while keeping the less-preferred providers warm as a fallback. This requires careful orchestration, because providers dislike discovering they are a stalking horse, but handled with discretion it allows the developer to secure its preferred outcome quickly while preserving competitive tension. The orchestration is itself a skill, and it is one that an experienced adviser frequently brings, which is part of the value such advisers add to a fast raise.

9. Due Diligence Compression

Due diligence is frequently the longest stage of a capital raise, as the provider conducts its commercial, financial, technical and legal investigations, and it is a stage where significant time can be saved through readiness and technology. Figure 6 compares traditional and technology-assisted diligence timelines.

Figure 6. Due Diligence Duration: Traditional versus Technology-Assisted



Technology-assisted review compresses each diligence workstream. Not a forecast.

The figure shows technology-assisted diligence compressing each workstream materially. The compression comes from two sources. The first is readiness: a complete, well-organised data room allows the provider diligence teams to find what they need quickly, without repeated requests to the developer, and a developer that has anticipated the diligence questions and addressed them in the data room removes the back-and-forth that consumes most diligence time. The second is technology: artificial intelligence tools that can review and extract from large volumes of documents, reconcile financial data, and flag anomalies allow diligence teams to cover more ground faster, compressing the commercial, financial and legal reviews in particular.

The developer can accelerate diligence by conducting its own diligence in advance and presenting the results, a practice known as vendor or sell-side diligence. A developer that commissions its own commercial, financial and technical diligence before going to market, and presents the reports in the data room, gives providers a head start and reduces the work they must duplicate, compressing the diligence stage. Vendor diligence also surfaces and allows the developer to address any issues before the provider finds them, avoiding the delay and loss of confidence that a provider discovery of an unaddressed issue causes. The cost of vendor diligence is repaid in the time it saves and the confidence it builds.

The role of artificial intelligence in diligence is growing rapidly and deserves emphasis, because it is changing what is achievable. Tools that can read and summarise contracts, reconcile financial models, extract key terms from leases and title documents, and flag inconsistencies allow both the developer and the provider to compress the document-intensive parts of diligence dramatically. A developer that uses these tools to prepare and quality-check its own data room, and that presents materials in a form these tools can readily process, both speeds its own preparation and accelerates the provider review. Section 16 returns to the technology architecture that supports an accelerated raise, of which AI-assisted diligence is a central component.

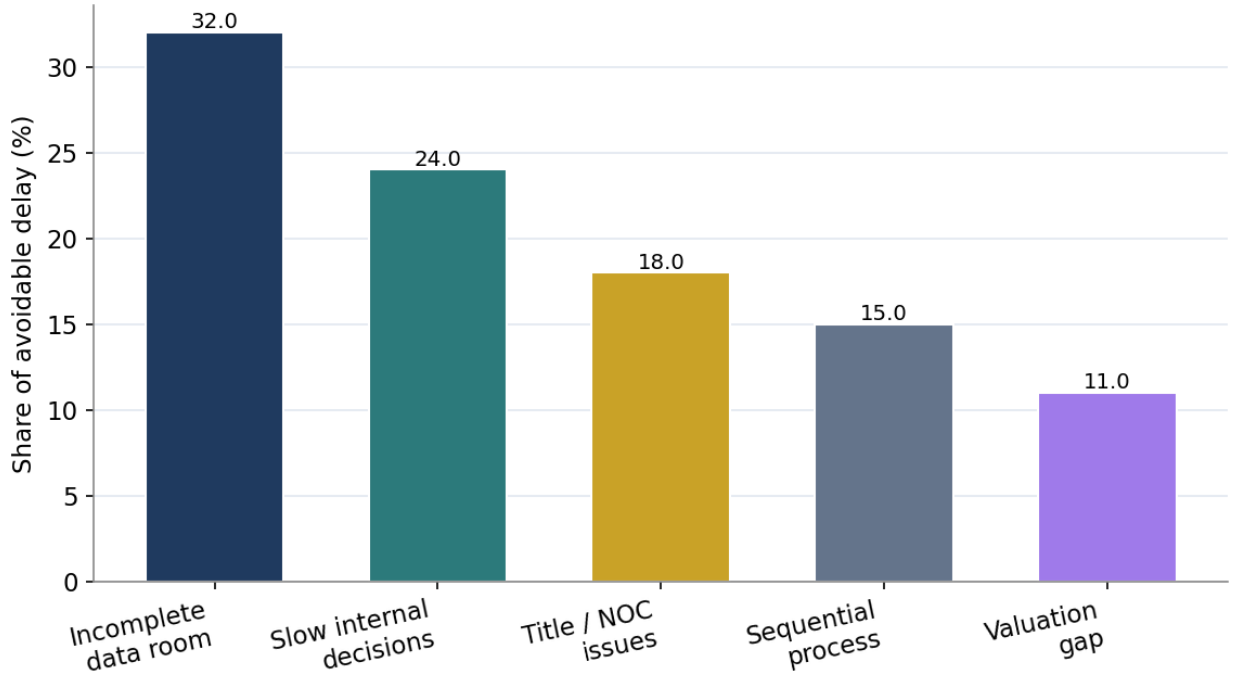
Vendor diligence merits a fuller defence because developers often resist its cost. The objection is that commissioning diligence the provider will in any case repeat seems wasteful. But the vendor

diligence is not wasted: it compresses the provider review by giving it a credible starting point, it surfaces issues while the developer still controls the narrative and can address them, and it signals confidence and preparation that build provider trust. The cost, modest relative to the size of most raises, is repaid in the time saved and the problems avoided, and a developer that has once lost a deal to a diligence surprise it could have pre-empted rarely objects to vendor diligence again. The practice is standard in mature markets precisely because its value is well understood.

10. Common Causes of Delay

Diagnosing where delay comes from is the prerequisite to removing it, and Figure 7 sets out the principal avoidable causes of delay in a capital raise, by their share of the avoidable delay.

Figure 7. Principal Avoidable Causes of Delay in a Capital Raise



The largest single cause is the incomplete data room, which accounts for roughly a third of avoidable delay, because every gap generates a question, a request and a wait. The second is slow internal decision-making by the developer, where the time taken to respond to a provider, to approve a term, or to make a decision adds up across the process; a developer with unclear internal authority and slow governance delays its own raise. The third is title and approval issues, where unresolved questions about the title, the registration or the necessary approvals stall the provider diligence and the documentation. The fourth is the sequential process discussed earlier, and the fifth is the valuation gap.

What these causes have in common is that they are all within the developer control. The provider response time, the part the developer cannot control, does not appear among the principal causes of avoidable delay, because providers presented with complete materials respond reasonably quickly. The implication is empowering: because the principal causes of delay are self-inflicted, the developer can largely eliminate them through its own preparation and discipline, without needing the providers to change their behaviour. The framework in this paper is, in essence, a structured attack on these self-inflicted causes.

Slow internal decision-making deserves particular attention because it is the least visible and the most cultural of the causes. A developer whose internal decision-making is slow, because authority is unclear, because decisions require many approvals, or because the principals are not engaged in the raise, will delay its own process regardless of how complete its data room is. The remedy is organisational: clear decision-making authority for the raise, a small empowered team, and the genuine engagement of the principals, so that the developer can respond to providers and make decisions at the speed the process demands. A fast raise requires a fast organisation, and a developer that is slow internally cannot run a fast raise no matter how well prepared its documents are.

11. Considerations Specific to the UAE

11.1 Escrow set-up and the regulatory path

A UAE development raise involves regulatory steps that bear on the timetable, most notably the establishment of the escrow account and the associated regulatory registrations for an off-plan project. These steps take time and must be planned into the timetable rather than discovered late, and a developer that initiates them early, in parallel with the capital raise, avoids the delay of completing them sequentially after the financing is agreed. The escrow and regulatory path is largely predictable, which means it can be planned and parallelised, and a developer that treats it as a known workstream to be run alongside the raise rather than a surprise to be handled afterward keeps it off the critical path.

11.2 Title, registration and no-objection certificates

The clarity and completeness of the title, the registration of the land and the development, and the necessary no-objection certificates from master developers and authorities are frequently the documents that, if incomplete, cause the most delay. A developer that ensures these are in order before launching the raise removes a major source of diligence delay, while a developer that launches with unresolved title or approval questions invites the provider to wait while they are resolved. Because these documents depend on third parties, the master developer, the authorities, the land department, they cannot be produced quickly under pressure, which is precisely why they must be secured in advance.

11.3 The depth and speed of the UAE capital market

The UAE benefits from a deep and increasingly competitive capital market, with banks, private credit funds, family offices and institutional investors all active, and from the dirham peg that allows international dollar capital to participate without currency friction. This depth is an advantage for speed, because a developer running a parallel process has many credible providers to approach and can generate genuine competition. The developer that understands and maintains relationships across this market can launch a parallel process quickly, drawing on established relationships rather than building them from scratch, which is itself a form of readiness that compresses the outreach stage.

Slow internal decision-making frequently traces to the absence of a single empowered owner of the raise. When responsibility is diffused across the principals, the finance function and external advisers, with no one clearly accountable for the timeline, decisions wait for consensus and the process drifts. The remedy is to appoint a single owner with the authority to drive the raise, make decisions within agreed parameters, and escalate only the genuinely material questions to the

principals. This owner functions as the developer counterpart to the provider deal lead, and the existence of empowered leads on both sides is one of the clearest predictors of a fast raise. Their absence is one of the clearest predictors of a slow one.

12. The Capital Provider Perspective

A raise moves quickly only if the provider can move quickly, and understanding what allows a provider to say yes fast is part of designing a fast raise. A provider can move quickly when it has everything it needs to form a view: a complete, consistent data room, a credible sponsor with a track record, clear security, and a deal that fits its mandate and risk appetite. A provider is forced to move slowly when it must chase missing information, reconcile inconsistencies, investigate an unfamiliar sponsor, or stretch its mandate to fit the deal. The developer that presents the former gets a fast yes; the developer that presents the latter gets a slow maybe.

Providers also move faster for deals and sponsors they find attractive, which means the developer reputation and the quality of the opportunity affect the speed as well as the terms. A provider presented with an attractive deal from a credible, prepared sponsor prioritises it, allocates its best people, and moves quickly to secure it against competition; a provider presented with a marginal deal from an unknown or disorganised sponsor deprioritises it, and it drifts. The developer that builds a reputation for quality and reliability, and that brings well-prepared attractive opportunities, finds that providers move quickly for it, which is a compounding advantage that preparation and track record build over time.

Matching the deal to the right provider is also central to speed, because a deal presented to a provider whose mandate it fits converts quickly, while a deal presented to a provider who must stretch to accommodate it converts slowly or not at all. A developer that understands the mandates, appetites and processes of the providers it approaches can target those most likely to move quickly on its particular deal, avoiding the wasted time of pursuing providers who were never a good fit. This targeting is part of the preparation: the developer that has mapped the provider universe and understands who wants what can direct its parallel process to the providers most likely to deliver a fast term sheet.

13. Indicative Case Studies

Three indicative cases show the framework in action. The figures are synthetic and constructed for analytical clarity, not drawn from any specific transaction.

13.1 Case A: the fast raise

Case A is a developer that prepares thoroughly before launch, with a complete data room, a reconciled model, clear title and pre-cleared approvals, and that runs a focused parallel process of well-targeted providers. It reaches a signed term sheet in ninety-five days and funds in around one hundred and fifty, capturing the acquisition opportunity that prompted the raise. The case illustrates the benchmark achieved: readiness and parallelism combine to deliver a term sheet within the target window, and the developer moves on its opportunity while slower competitors are still preparing.

Table 6. Case A Timeline, the Fast Raise

Stage	Days	Cumulative	Note

Preparation	Pre-launch	0	Completed before launch
Outreach	30	30	Parallel, well-targeted
Term sheet	35	65	Competitive tension
Diligence (overlapped)	30	95	Vendor diligence provided
Signed term sheet		**95**	Within benchmark

Readiness and parallelism deliver a term sheet in 95 days. Not transaction-specific.

13.2 Case B: the delayed raise

Case B is a developer that launches with an incomplete data room, unresolved title questions and a sequential process, approaching one provider at a time. It reaches a term sheet only after two hundred and sixty days, by which time the acquisition it had hoped to fund has been lost to a faster competitor, and it funds after three hundred and sixty days into a less attractive alternative. The case illustrates the cost of unreadiness: the self-inflicted delays compound, the opportunity slips, and the developer ends up worse off despite, in the end, raising the capital. The contrast with Case A is entirely a matter of preparation and process, not of market conditions.

Table 7. Case B Timeline, the Delayed Raise

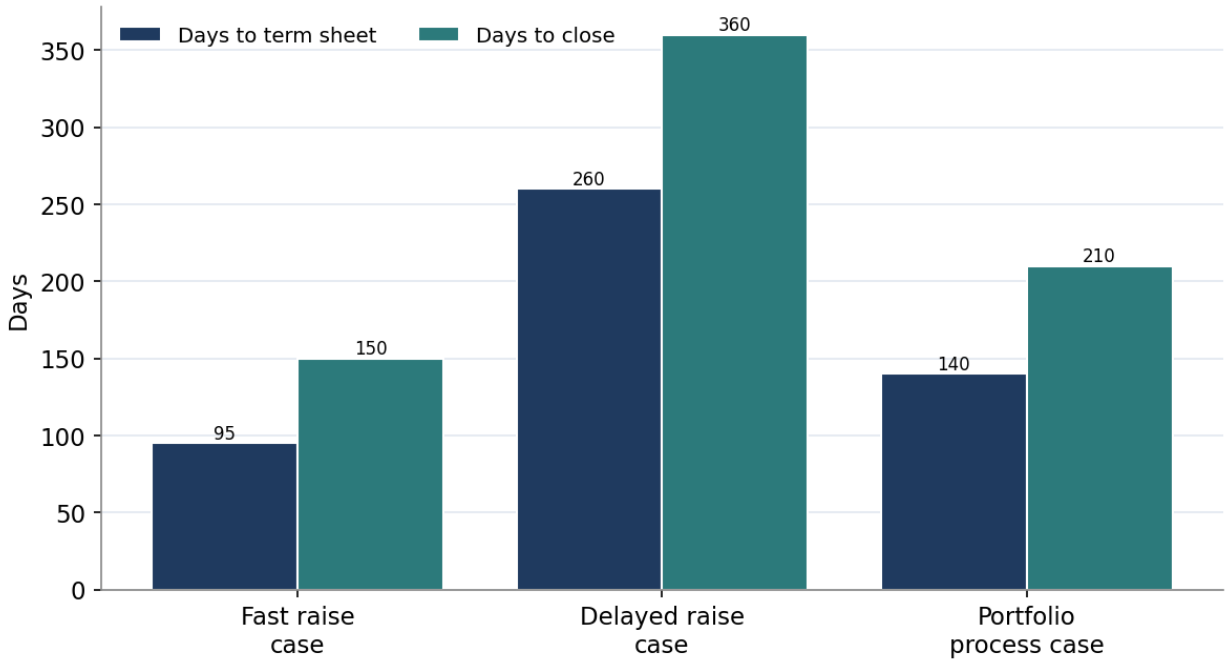
Stage	Days	Cumulative	Cause of delay
Preparation (reactive)	60	60	Data room assembled after launch
Outreach (sequential)	70	130	One provider at a time
Term sheet	60	190	Valuation gap surfaced late
Diligence	70	260	Title questions unresolved
Signed term sheet		**260**	Opportunity lost by now

Self-inflicted delays compound to 260 days. Not transaction-specific.

13.3 Case C: the portfolio process

Case C is a developer that institutionalises its capital raising, maintaining a standing data room, established provider relationships and a repeatable process across a portfolio of projects. It reaches term sheets in around one hundred and forty days even on complex portfolio raises, because the standing readiness and relationships mean each raise begins from a position of preparation rather than from scratch. The case illustrates the strategic endpoint of the framework: a developer that treats speed as an institutional capability, embedded in standing processes and relationships, rather than as something improvised for each raise.

Figure 8. Days to Term Sheet and to Close by Case



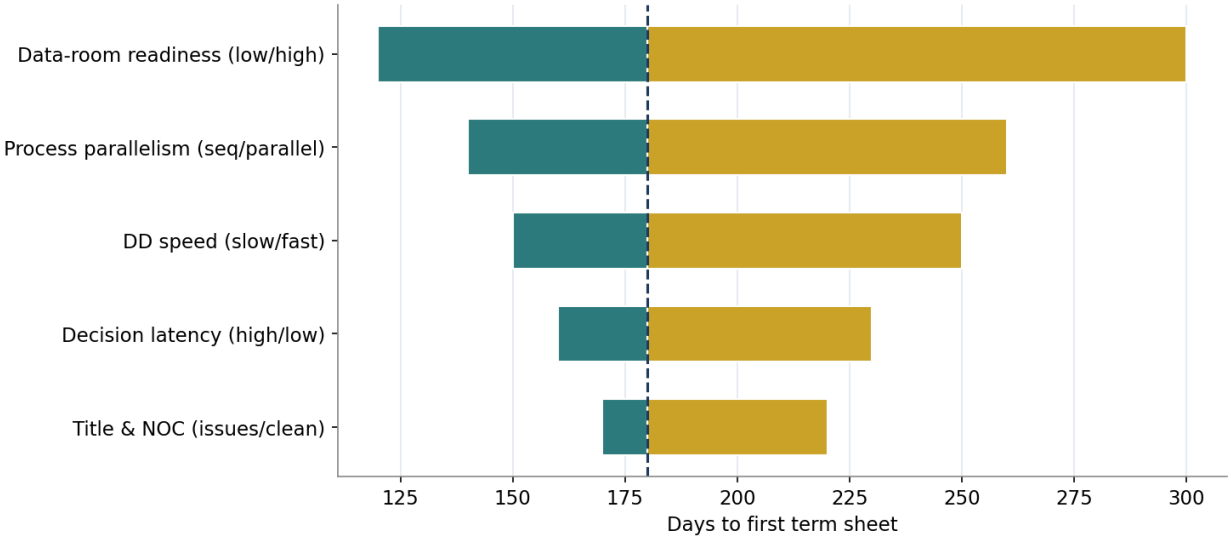
Synthetic figures for analytical comparison. Not a forecast.

Figure 8 compares the three cases on the days to term sheet and to close. The fast raise and the portfolio process reach term sheets within or near the benchmark window, while the delayed raise takes nearly three times as long. The comparison underlines that the difference between fast and slow raises is not market conditions, which were similar across the cases, but the developer preparation, process design and institutional capability. The framework is, in effect, a recipe for moving a developer from the Case B profile to the Case A and Case C profiles, and the prize is the difference in time, cost and opportunity capture between them.

14. Sensitivity and Scenario Analysis

A tornado analysis identifies the variables that most influence the days to a first term sheet. Figure 9 presents the result.

Figure 9. Sensitivity of Days to Term Sheet to Key Variables



Each bar shows the days to term sheet when the labelled variable moves to its low or high case. Dashed line is the base case. Indicative.

The analysis confirms the themes of the paper: data-room readiness and process parallelism dominate the time-to-capital, with diligence speed and decision latency also significant, while factors outside the developer control matter less. The variable with the largest single effect is data-room readiness, which can swing the time to term sheet from around one hundred and twenty days when readiness is high to three hundred when it is low. The implication is unambiguous: the developer that wishes to compress its time-to-capital should invest first and most in its data-room readiness, then in its process design, because these are the variables that move the outcome most and that are entirely within its control.

Table 2. Scenario Matrix for Days to First Term Sheet

Scenario	Readiness	Process	Days to term sheet
Best practice	High	Parallel	95
Typical	Moderate	Mixed	180
Poor	Low	Sequential	300
Distressed	Low	Sequential, weak position	420

Indicative scenarios. Not a forecast.

The scenario matrix makes the prize concrete. Moving from the typical to the best-practice profile saves roughly eighty-five days, and from the poor to the best-practice profile roughly two hundred days, on every raise. For a developer that raises capital several times a year, the cumulative saving in carrying cost, opportunity capture and reputational benefit is substantial, and it is achieved purely through preparation and process discipline, without any change in market conditions or any concession on the cost of capital. The scenario analysis is, in effect, a quantification of the return on investing in readiness, and that return is high.

15. International Comparison

The discipline of fast capital raising is well developed in the mature markets, where competitive processes, professional advisers and standardised diligence have compressed timelines and where developers and sponsors routinely run tightly managed parallel processes. In the United Kingdom and the United States, a well-advised sponsor with a prepared data room can reach a term sheet in a comparable ninety to one hundred and twenty day window, and the practices that achieve this, thorough preparation, vendor diligence, competitive parallel processes and disciplined timetables, are the same practices this paper recommends for the UAE.

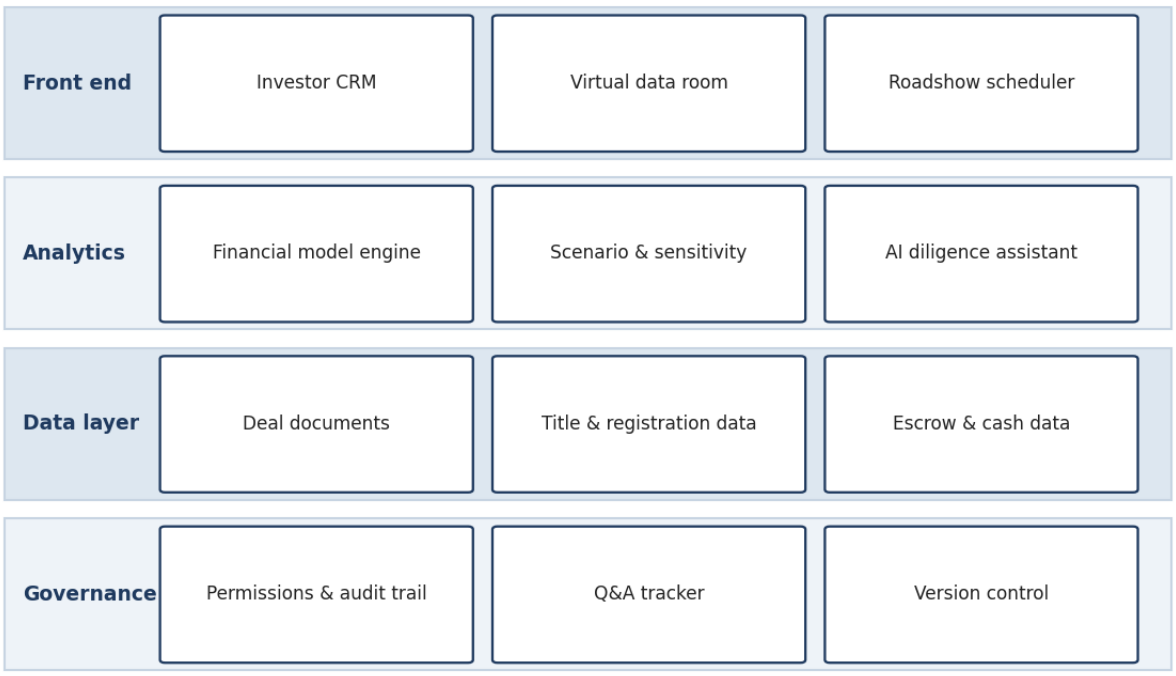
The lesson from the mature markets is that speed is a function of professionalism and standardisation, and that as a market matures its timelines compress. The UAE market is on this trajectory, with deepening capital, more professional advisers and increasingly standardised processes, and the developers that adopt the disciplines of the mature markets ahead of their peers gain a speed advantage while the market as a whole catches up. The international comparison also highlights the role of professional advisers, who bring established provider relationships, standardised processes and diligence experience that compress timelines, and a UAE developer that engages capable advisers can import this speed rather than building it from scratch.

The comparison with mature markets also cautions against importing their practices uncritically. The UAE has its own regulatory steps, its own escrow and registration requirements, and its own provider landscape, and a process designed for the United Kingdom or the United States must be adapted to these local realities rather than transplanted wholesale. The principles, readiness, parallelism, vendor diligence, disciplined timetables, travel intact, but their implementation must reflect the local regulatory path and the local provider expectations. The most effective approach combines the disciplines of the mature markets with a sophisticated understanding of the local process, which is precisely the combination that an experienced local adviser with international standards brings.

16. Technology and Artificial Intelligence in Compressing Time-to-Capital

Technology is increasingly central to compressing time-to-capital, and a developer that builds the right technology capability can run faster raises than one that relies on manual processes. Figure 10 sets out an indicative technology architecture for an accelerated capital raise.

Figure 10. Technology Architecture for an Accelerated Capital Raise



Indicative reference architecture. Not a specific product recommendation.

The architecture has four layers. The front end provides the investor relationship management, the virtual data room and the roadshow scheduling that organise the outreach and the provision of materials. The analytics layer provides the financial model engine, the scenario and sensitivity analysis, and the artificial intelligence diligence assistant that accelerate the analytical work. The data layer holds the deal documents, the title and registration data, and the escrow and cash data that the raise depends on. The governance layer provides the permissions, audit trail, question tracker and version control that keep a fast, parallel, multi-provider process organised and controlled. Together these layers allow a developer to run a fast raise without the chaos that speed can otherwise bring.

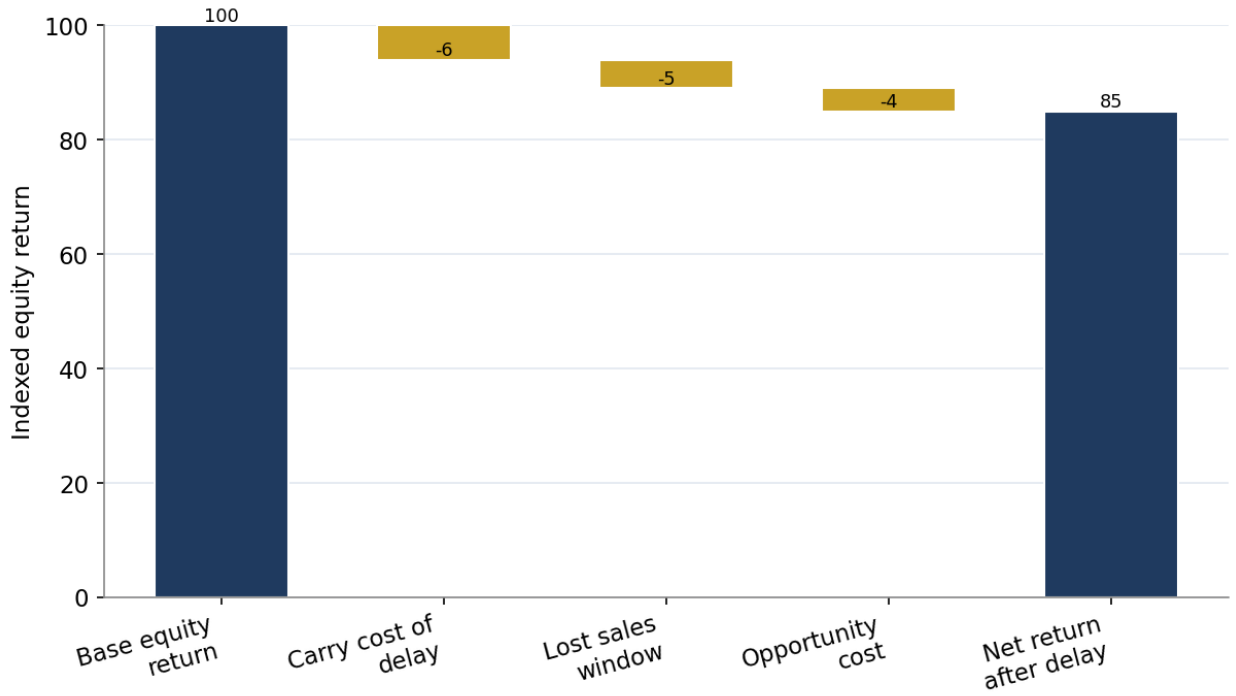
Artificial intelligence contributes across the architecture but most powerfully in the preparation and diligence of documents. AI tools can assemble and quality-check the data room, reconcile the model to the appraisal, extract and summarise the key terms of leases, contracts and title documents, and flag inconsistencies before a provider finds them, compressing both the developer preparation and the provider diligence. A developer that uses these tools to reach a higher state of readiness faster, and to present materials in a form that the provider own AI tools can readily process, accelerates the whole raise. The technology does not replace judgement, but it removes the document-handling drudgery that consumes much of the elapsed time, freeing the human effort for the genuine analysis and negotiation.

The investment in this technology capability is justified by its reuse across many raises. A developer that builds a standing data room, an investor relationship system and an AI-assisted diligence capability amortises the cost across every subsequent raise, each of which begins from a higher state of readiness and runs faster. This connects to the strategic perspective of Section 20: the technology is part of institutionalising speed as a capability, converting it from something improvised for each raise into a standing asset of the business. The developer that makes this investment runs faster raises not occasionally but systematically, which is the durable form of the advantage.

17. The Cost of Delay Quantified

To make the case for investing in speed concrete, it helps to quantify the cost of delay on a representative raise. Figure 11 sets out an indicative decomposition of how delay erodes the equity return.

Figure 11. Indicative Erosion of Equity Return Through Delay (Indexed)



Delay erodes the return through carry, lost sales window and opportunity cost. Not a forecast.

The decomposition shows three channels through which delay erodes the return. The carry cost of the delay accrues directly against the project. The lost sales window, where a delayed launch pushes sales into a less favourable period or shortens the time available to sell before a deadline,

reduces the realised revenue. And the opportunity cost, where the capital and management attention tied up in the delayed raise cannot be deployed elsewhere, forgoes the return that deployment would have earned. Together these channels can erode the equity return by a meaningful margin, and the erosion grows with the length of the delay.

Setting this cost of delay against the cost of investing in readiness and process makes the business case clear. The investment in a complete data room, a standing technology capability and a disciplined process is modest relative to the cost of delay it avoids, particularly when amortised across many raises. A developer that compares the one-time and recurring cost of building a fast-raise capability against the per-raise cost of delay it suffers without one will find the investment easily justified, often many times over. The cost of delay is, in effect, the return on investing in speed, and Figure 11 is a tool for making that return visible to a developer deciding whether to make the investment.

18. Common Errors and How to Avoid Them

A recognisable set of errors recurs in slow capital raises, each flowing from a failure to treat speed as something to be engineered.

- **Launching before ready.** Launching a raise before the data room is complete forces materials to be assembled reactively under pressure, the single largest cause of delay. The remedy is to complete and quality-check the data room before launch.
- **Running a sequential process.** Approaching providers one at a time forfeits competitive tension and runs the stages sequentially. The remedy is a focused parallel process of well-targeted providers.
- **Slow internal decisions.** Unclear internal authority and slow decision-making delay the developer own responses. The remedy is a small, empowered team with clear authority and engaged principals.
- **Unresolved title and approvals.** Leaving title, registration and approvals unresolved at launch invites diligence delay that cannot be fixed quickly because it depends on third parties. The remedy is to secure these documents in advance.
- **Mishandling exclusivity.** Granting long, unconditional exclusivity ties the developer to a single, unpressured provider. The remedy is short, conditional exclusivity tied to the timetable.

Each of these errors is avoidable, and each is within the developer control. The common thread is that speed must be engineered in advance, through preparation, process design and organisation, rather than hoped for once the raise is under way. The developer that engineers speed avoids these errors more or less automatically, because the discipline that produces speed is the same discipline that prevents the errors.

19. Implementation Roadmap

1. Build and quality-check the complete data room before launching, against a structured readiness checklist covering appraisal, model, title, approvals, cost, sales and governance materials.
2. Resolve title, registration and approval questions in advance, since these depend on third parties and cannot be fixed quickly under pressure.

3. Map and maintain relationships across the provider universe, so that a parallel process can be launched quickly from established relationships rather than built from scratch.
4. Design the process to run providers and stages in parallel, with a focused set of well-targeted providers, to create competitive tension and overlap the stages.
5. Establish clear internal decision-making authority and a small empowered team, so the developer can respond and decide at the speed the process demands.
6. Commission vendor diligence and use technology, including AI-assisted document review, to compress both preparation and provider diligence.
7. Manage the term sheet and exclusivity deliberately, coming to the negotiation with a clear prior position and keeping exclusivity short and conditional on the timetable.

The cost-of-delay analysis can be turned into a simple internal policy: a developer can set a target time-to-term-sheet, measure its performance against it, and treat any shortfall as a problem to be diagnosed and fixed, much as a manufacturer treats a production delay. This converts speed from an aspiration into a managed metric, with an owner, a target and a feedback loop. Developers that manage time-to-capital in this way improve it systematically over time, because what is measured and owned tends to improve, while developers that leave it unmeasured improve only by accident. The discipline of measurement is the bridge between recognising that speed matters and actually achieving it.

20. Strategic Perspective: Institutionalising Speed

The deepest lesson of this paper is that speed should be institutionalised as a standing capability rather than improvised for each raise. A developer that treats each raise as a fresh, ad hoc effort, assembling materials, building relationships and designing a process from scratch each time, will be slow each time, no matter how hard it works on the individual raise. A developer that builds standing capabilities, a maintained data room, established provider relationships, a repeatable process, a technology platform and an experienced team, begins each raise from a position of readiness and runs each raise fast, because the work of preparation has already been done.

Institutionalising speed turns it from a project into a property of the business. The developer that has done so can respond to opportunities as they arise, can run competitive parallel processes as a matter of routine, and can reach term sheets within the benchmark window consistently rather than occasionally. This consistency is itself valuable, because it allows the developer to commit to opportunities with confidence in its ability to fund them, and it builds the reputation for reliable, fast execution that causes providers to prioritise the developer and bring it opportunities first. Speed, institutionalised, becomes a compounding strategic advantage.

The investment required to institutionalise speed is modest relative to its return, and it is the kind of investment that distinguishes a developer building a durable business from one executing a series of one-off deals. The maintained data room, the provider relationships, the technology platform and the experienced team are assets that appreciate with use, each raise adding to the relationships and refining the process, so that the developer becomes faster over time. The developer that makes this investment early compounds the advantage; the developer that improvises each raise never builds it. For a developer with ambitions to grow through repeated capital raising, institutionalising speed is among the highest-return investments available.

21. Conclusion

Time-to-capital is a competitive variable that UAE developers can and should manage deliberately. This paper has argued that a prepared developer can reach a signed term sheet within ninety to one hundred and twenty days, that the gap between this benchmark and the many months an unprepared developer takes is largely self-inflicted, and that closing it is one of the highest-return process improvements a developer can make. The dominant levers are data-room readiness and parallel process design, both entirely within the developer control, and the cost of delay they avoid, in carrying cost, lost opportunities and reputational damage, is real and quantifiable.

The developer that internalises these lessons, completing its data room before launch, resolving its approvals in advance, running a focused parallel process, deciding quickly internally, and using technology to compress preparation and diligence, will consistently reach term sheets within the benchmark window and capture the opportunities that slower competitors lose. The developer that goes further and institutionalises speed as a standing capability, embedded in maintained data rooms, provider relationships, technology and an experienced team, converts the advantage from occasional to systematic. In a competitive UAE market where opportunities are time-limited and capital is plentiful but selective, the ability to move fast is a durable source of advantage, and the frameworks and tools in this paper are intended to help developers build it.

22. Limitations and Directions for Further Research

This paper is framework-oriented and relies on indicative data, and its conclusions are directional rather than precise. The benchmark timelines, readiness relationships and cost-of-delay figures are calibrated to observable conditions but are not empirical estimates drawn from a dataset of completed raises, and they vary with deal size, complexity and market conditions. The technology and artificial intelligence capabilities described are evolving rapidly, and the specific tools available will change.

Several extensions would strengthen the analysis. An empirical study of time-to-term-sheet across a sample of UAE development raises, correlated with measures of readiness and process design, would replace the indicative relationships with data. A study of the realised cost of delay across completed and lost transactions would quantify the business case more precisely. And an examination of how artificial intelligence tools are changing diligence timelines in practice would sharpen the technology dimension. Each is a natural subject for a later paper in this series.

Appendix A. Capital-Raise Readiness Checklist

Table 8. Indicative Readiness Checklist

Category	Key items
Appraisal and model	Development appraisal; financial model reconciled to appraisal; scenarios
Title and registration	Title documents; land registration; ownership structure; encumbrances
Approvals	Zoning and planning approvals; master-developer NOCs; authority consents
Cost and programme	Cost plan; construction programme; contractor and consultant details
Sales and market	Sales evidence; price list; absorption assumptions; market study

Corporate and governance	Corporate structure; board approvals; KYC; prior accounts
Financing	Existing facilities; security; escrow status; sources and uses
Diligence	Vendor commercial, financial and technical diligence reports

Indicative checklist. Completeness, consistency and credibility across these items drive the readiness score. Not exhaustive.

Appendix B. Base Case Assumptions

Table 9. Base Case Assumptions

Parameter	Value	Parameter	Value
Benchmark to term sheet	90-120 days	Term sheet to close	60-90 days
Top-quartile days	~105	Median days	~180
Readiness score (best practice)	>80	Parallel providers	3-5
DD compression with technology	~30-40%	Cost of delay	~7% of return at 120 days
Currency	AED (USD peg)	Process maturity target	Level 4-5

Indicative parameters used to generate the figures and case studies. Not transaction-specific.

Appendix C. Glossary of Terms

Table 10. Glossary of Key Terms

Term	Definition
Time-to-capital	Elapsed time from launching a raise to securing capital.
Term sheet	Heads of terms setting out the principal commercial terms, subject to diligence.
Data room	The organised set of documents provided to capital providers.
Readiness	The completeness, consistency and credibility of materials at launch.
Parallel process	Approaching multiple providers and overlapping stages simultaneously.
Vendor diligence	Sell-side diligence commissioned by the developer in advance.
Exclusivity	A period during which the developer deals with one provider only.
Conditions precedent	Conditions that must be satisfied before funding.
Cost of delay	The carrying, opportunity and slippage cost of a slow raise.
Process maturity	The degree to which fast raising is institutionalised.

Definitions provided for reference.

References

[1] Brealey R., Myers S., Allen F., Principles of Corporate Finance, 13th Edition, McGraw-Hill Education, 2020.

[2] Rosenbaum J., Pearl J., Investment Banking: Valuation, LBOs, M&A, and IPOs, 3rd Edition, Wiley, 2020.

[3] Geltner D., Miller N., Clayton J., Eichholtz P., Commercial Real Estate Analysis and Investments, 3rd Edition, OnCourse Learning, 2014.

[4] Reinertsen D., The Principles of Product Development Flow, Celeritas Publishing, 2009.

[5] Preqin, Private Capital Fundraising and Deal Timelines, Market Overview, 2024.

[6] Damodaran A., Applied Corporate Finance, 4th Edition, Wiley, 2015.

[7] Royal Institution of Chartered Surveyors, Development Appraisal and Valuation, RICS Guidance, 2019.

[8] World Economic Forum, Artificial Intelligence in Financial Services, WEF, 2023.

[9] Regulatory frameworks for escrow and off-plan registration, Dubai Land Department and Real Estate Regulatory Agency public materials, accessed 2026.

[10] Bank for International Settlements, Digitalisation of Finance, BIS Papers, 2021.

About the Author

Chennakeshav Adya is an independent researcher and corporate finance practitioner with more than twenty years of international experience across business strategy, transformation, and strategic finance. He has originated and led transactions across real estate, technology, and clean energy in the UAE, the United Kingdom, India, Europe, and the Americas, and has advised developers, corporates, family offices, and fund managers on capital raising and structuring. He is a guest speaker at London Business School and writes on startups, capital markets, and the application of technology in finance. The views expressed in this paper are his own and do not constitute investment advice.