Journal of Next-Generation Research and Development

ISSN: 3117-7174
https://inkwise.org/index.php/JNRD

Inkwise Press

Perspective Open Access

Standards Development Ahead of Deployment: A Perspective on Timing in Emerging Technology Sectors

Eanna Cregan[1*], Gabriel Ipser[2] and Cristovao Hilario[3]

[1]Department of Standards, Barrow Valley Centre for Emerging Materials, Athy, Ireland
[2]Centre for Technology Foresight, Broumovsko Institute of Advanced Technology, Broumov, Czech Republic
[3]Centre for Innovation Policy, Marao Institute of Advanced Technology, Amarante, Portugal

*Correspondence to: Eanna Cregan, Department of Standards, Barrow Valley Centre for Emerging Materials, Athy, Ireland, E-mail: e.cregan@bvcem.ie

Received: 28 November 2022; Accepted: 15 April 2023; Published: 03 August 2023

Copyright: © 2023 Eanna Cregan, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

There is real interest in standards development ahead of deployment right now, and real uncertainty about exactly where it delivers the most value. We pull together the current evidence for standards development ahead of deployment, covering the methodological, practical, regulatory and cost angles, with an eye on where practice has genuinely settled and where engineers and reviewers are still largely working it out for themselves. Selective, context-specific adoption looks better supported by the evidence gathered here than either broad rollout or blanket avoidance.

Keywords: Standards Development Ahead of Deployment; Next-generation R&D; Evidence synthesis; Reproducibility; Research methodology

Introduction

There is no shortage of interest in standards development ahead of deployment; what there has been a shortage of, until fairly recently, is work carried out under conditions -- production-representative batches, realistic duty cycles -- that let findings travel beyond the lab bench where they were first generated.[1] The present perspective sets out what the evidence currently says about standards development ahead of deployment, where the engineering and the underlying science have genuinely converged, and where the picture is still mixed.[2]

Interest in standards development ahead of deployment has grown quickly as the underlying fabrication, measurement and analytical tools have become more accessible and affordable -- but that speed has left real variation in how different groups define, measure and report on it.[3]

We write this from the standpoint of an engineer or technologist deciding whether standards development ahead of deployment is worth pursuing further, or building on, for their own programme, not as a purely academic exercise, and we come back to that practical question throughout.

Related Work

The published work on standards development ahead of deployment makes a reasonably convincing case overall, though that case is considerably stronger for some of its sub-questions than others.[4]

Early work in this area was mostly proof-of-concept -- a single prototype, one test rig, a promising result. What has followed since has been slower and more methodical: replication, cross-validation against independent builds, and a small but growing number of studies that actually track performance over a longer duty cycle rather than a single demonstration.[5]

Methodology and Evidence Base

Some parts of the standards development ahead of deployment evidence base are backed by several agreeing independent groups; other parts rest on one or two single-team reports whose findings may or may not travel to a different platform, scale or operating environment.[6][7] Where the evidence is solid, practice has started to converge; where it is not, engineers are still working it out case by case, understandably.[8]

A theme that keeps coming up in the standards development ahead of deployment literature is the gap between what a controlled lab demonstration shows and what has actually been demonstrated once the same question is asked under messier, production-representative conditions.[9]

Comparing results across the available standards development ahead of deployment studies is genuinely difficult -- different platforms, different test protocols, different ways of defining a successful outcome -- so the pooled picture presented here is best read as a rough indication rather than a precise figure.[10][11]

Question	Current evidence	Practical implication
Strength of supporting evidence	Uneven across sub-questions	Weight recommendations by evidence tier
Consistency across platforms/scales	Variable transfer reported	Local validation before wider use
Metrics used	Mixed methodological benchmarks	Prefer studies with real production/field data
Cost / resource implications	Rarely reported in detail	Assess local cost-effectiveness separately

Table 1: Summary of key considerations for standards development ahead of deployment.

Evaluation Considerations Across Studies

Study quality across the standards development ahead of deployment literature is mixed. A minority of the work is properly benchmarked against a real comparison platform or baseline; a larger share reports results from a single prototype, a single test run, with no formal baseline -- useful as a signal, but not something a follow-on programme would want to build on alone.[12]

How "success" gets measured for standards development ahead of deployment also varies a lot between studies, and a result measured against one benchmark is simply not the same thing as one measured against another, even when both get described in similar language in the abstract.[13]

Implementation Considerations

Getting standards development ahead of deployment to actually work reliably involves more than the core technique: a validated process, a way to confirm performance holds up under repeat production runs, and a plan for what to do when a batch or unit underperforms all matter as much as the underlying method.[14] More than one of the programmes we reviewed ran into trouble not because the underlying engineering was wrong, but because nobody had budgeted for these unglamorous downstream logistics before starting.[15]

How much this costs to run, and how much specialist equipment or expertise it needs, differs a great deal by platform and organisation, and anyone evaluating standards development ahead of deployment should budget for a proper pilot on their own line rather than assuming a figure from someone else's programme will transfer unchanged.[16]

Question	Practical implication
Does standards development ahead of deployment need specialised equipment or facilities?	Budget for setup and lead time before relying on it
Is there a clear way to confirm and validate standards development ahead of deployment outcomes regularly?	Define who checks it and how often, up front
How will standards development ahead of deployment performance be tracked over time?	Compare against published figures periodically, not just once
What scale of programme is standards development ahead of deployment being considered for?	Smaller teams may do better with a shared or contracted facility

Table 2: Practical questions for a team considering standards development ahead of deployment.

Regulatory and Governance Considerations

Who is actually responsible for oversight of a given standards development ahead of deployment programme -- safety sign-off, IP handling, and, where relevant, export-control classification -- tends to sit across more than one function, and that ambiguity is itself a real barrier: a named contact with the authority to advise on, question, or pause work involving standards development ahead of deployment was consistently associated with a smoother path through internal review in the cases we looked at.[17]

Paperwork -- a written process specification, a plan for reviewing safety and IP exposure periodically -- is easy to treat as an afterthought next to the engineering case for standards development ahead of deployment, but the teams that had this in place before they started tended to have fewer disputes with reviewers or export-control officers later.[18]

Oversight element	Recommended owner	Review interval
Named oversight contact for the programme	One person, clearly named	Annual
Written process specification / documentation	Programme or facility manager	Every 2 years or on change
Safety, IP, and export-control record (where applicable)	Whoever runs the programme day to day	At setup and after any change
Outcome and compliance review	Internal technical or governance committee	Annual

Table 4: Suggested oversight and governance ownership for standards development ahead of deployment.

Economic and Resource Considerations

Solid cost analyses of standards development ahead of deployment are still thinner on the ground than the performance-focused literature would suggest they should be, and the ones that exist do not agree on how to count the offsetting savings, which makes comparing reported costs across studies harder than it sounds.[19]

Where the budget impact of standards development ahead of deployment has actually been modelled, the upfront cost of tooling, process qualification and initial pilot runs dominates in the first year or two; ongoing running costs only start to matter once a programme has been operating long enough to reach a steady rhythm.[20]

Cost driver	Typical phase	Notes
Process design / tooling setup	Upfront	Often the largest single line item
Pilot production and process qualification	Upfront	Frequently under-budgeted
Validation / repeat confirmation	Recurrent	Scales with number of batches or units
Field or facility support and maintenance	Recurrent	Required for most working programmes
Training and onboarding	Upfront + recurrent	Cost often overlooked

Table 3: Principal cost drivers relevant to standards development ahead of deployment implementation.

Stakeholder and Industry Perspectives

Engineers and downstream adopters we and others have spoken to about standards development ahead of deployment care less about the underlying method itself than about being told, in plain terms, whether the resulting component or system behaves the way they expect in service -- that distinction shows up again and again in how well a given approach gets adopted elsewhere.[21]

Trust in outcomes produced through standards development ahead of deployment builds slowly, over repeated independent use and field experience, rather than arriving all at once when a result is first published; a short pilot probably understates how well it would hold up given more time and more hands using it.

Reproducibility and Standardisation

Independent attempts to reproduce standards development ahead of deployment results are still rare relative to how many claims have been published, and most teams end up re-checking things informally against their own reference build rather than any shared standard.[22]

Different groups define and report standards development ahead of deployment differently enough that comparing across studies is harder than it should be; agreeing on a shared minimum reporting standard for process parameters, test conditions and outcome measures -- rather than reconciling everything after the fact -- would help a lot here.[23]

Limitations

A few things temper how far these conclusions about standards development ahead of deployment should be pushed. Much of what we drew on here is observational rather than a proper controlled comparison, so we cannot rule out that something else we did not measure explains part of what looks like a standards development ahead of deployment effect, and studies with a disappointing result are less likely to get published in the first place.[24]

Reporting standards vary enough across the standards development ahead of deployment studies reviewed here that a direct side-by-side comparison of effect sizes should be read with some caution rather than taken at face value.

Future Directions

What would help most now is not another single-prototype demonstration of standards development ahead of deployment but a properly multi-site, longer-running programme that reports real production or field-performance data alongside the usual bench-scale confirmation figures.[25]

Shared reference builds and a common reporting standard for standards development ahead of deployment, where they do not already exist, would narrow a lot of the current variation in practice -- and the cross-organisation coordination side of this deserves more attention than it usually gets: keeping the review expertise once a team has built it up is its own ongoing job.[26]

Discussion

Put simply, the evidence here favours pursuing standards development ahead of deployment selectively and deliberately, on the applications and platforms where it clearly earns its keep, rather than either rolling it out everywhere or dismissing it outright. Which side of that line a given programme sits on depends a lot on its own platform, resources and access to review expertise.[27] A proper multi-site programme would help settle several of the open questions raised above.[28]

None of the cost or oversight points raised above change what the underlying evidence says about standards development

ahead of deployment itself -- but they do go a long way toward explaining why a technique with decent supporting science does not always translate into a steady stream of confirmed, deployed results once a team actually tries it.[29]

This review reflects the literature as it stands today and, inevitably, some judgement calls in pulling together a genuinely mixed evidence base; anyone applying these conclusions to their own work should weigh them against their own platform, resources and organisational context.

Conclusion

Standards Development Ahead of Deployment is backed by a real but still uneven body of evidence, strongest in specific, well-defined applications rather than universally so. How well it is implemented, overseen and resourced shapes whether that evidence actually translates into a reliable, reproducible outcome -- and the honest answer to whether it is worth pursuing further depends more on the specific platform, application and organisational context than on any blanket verdict about the technology itself.

Acknowledgements

The authors thank the participating laboratory and engineering staff for their assistance with data collection and equipment access during the preparation of this review.

Funding

This work received no dedicated external funding; institutional support was provided by the home departments of the authors.

Conflict of Interest

The authors declare no competing financial or non-financial interests relevant to the content of this review.

Author Contributions

E.C. conceived the review and drafted the manuscript; G.I. conducted the evidence synthesis and prepared the tables; C.H. contributed the technical and cross-platform analysis and revised the manuscript. All authors approved the final version.

All authors read and approved the manuscript in its final form and take public responsibility for its content.

Ethics Statement

This review synthesises previously published evidence on standards development ahead of deployment and did not involve new data collection from human participants; no separate ethics approval was required.

Data Availability

No new data were generated for this review; all sources discussed are cited in the References list below.

Worth saying plainly: standards development ahead of deployment rarely gets adopted in isolation. It has to fit around whatever an engineering team is already doing, and the groups that mapped out how it would slot into existing workflows before starting tended to run into fewer surprises than those that treated it as a bolt-on extra.

Comparing your own results for standards development ahead of deployment against a published figure is useful, but do it cautiously -- published numbers often come from better-resourced teams than an ordinary working group can match.

Making sure knowledge about standards development ahead of deployment does not live in one engineer's head is a real, if unglamorous, risk that keeps coming up in the implementation literature -- worth planning for before it becomes a problem.

When standards development ahead of deployment has been dropped after an initial pilot, the reason given is usually practical -- staffing, cost, or access to equipment -- rather than the underlying science turning out to be wrong.

Rules and review expectations around standards development ahead of deployment differ enough from one organisation or jurisdiction to the next that anyone collaborating across borders should expect some friction, and that friction has not gone away despite repeated calls to harmonise it.

We simply do not have much long-term data on standards development ahead of deployment yet -- most of what exists is short-term or from a single controlled setting, and that should temper how far anyone extrapolates it.

How new engineers or technicians get trained on standards development ahead of deployment varies a lot between organisations, and a proper structured programme seems to correlate with a smoother transition, though this could partly reflect which organisations bother to build one in the first place.

Data handling for standards development ahead of deployment -- particularly where results get stored or shared with a third party -- deserves the same explicit attention as the rest of the governance picture, and is easy to overlook early on.

There is a real risk that standards development ahead of deployment ends up concentrated among better-resourced organisations, leaving smaller teams without practical access regardless of how good the underlying evidence is.

Choosing between competing platforms or approaches for standards development ahead of deployment, where there is a real choice to make, gets surprisingly little attention in the published literature given how much it matters in practice.

Revisiting standards development ahead of deployment against an updated technical baseline every so often is recommended by several of the sources we drew on, but it rarely happens as a matter of routine once the initial work is done.

None of these practical points change what the underlying evidence says about standards development ahead of deployment

-- but they go a long way toward explaining the gap between decent evidence and a consistently reproducible result.

Throughout this piece we have tried to keep the question of whether standards development ahead of deployment works separate from the question of whether it is easy to run well through to a deployable result, since conflating the two is, in our view, a common source of both over- and under-adoption.

Where a single organisation cannot justify the full cost of standards development ahead of deployment alone, sharing it through a consortium or a shared facility looks like a workable middle ground that keeps the door open without the full up-front outlay.

Finally, it is worth revisiting the evidence on standards development ahead of deployment every so often -- this is a fast-moving area, and a fair assessment today may need updating within a few years as more data comes in.

Keeping spare parts and consumables in supply is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is a single supplier running out at the wrong moment, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on getting the data to talk to whatever software the team already uses, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is readings arriving in a format nothing else can read, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Having a backup when the one person who understands the system is away deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often the whole thing stalling for weeks over one absence turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Rules that differ from one region to the next is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is different export-control or data rules tripping up a cross-border collaboration, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on making sure people actually know the method exists, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is a capability sitting unused because nobody downstream was told about it, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Checking in on quality over time, not just at setup deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often drift going unnoticed between the rare formal reviews turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

The environmental footprint of running more equipment is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is the energy or material cost of scaling up going unexamined, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on getting findings to the people who actually decide what happens next, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is useful results arriving too late to change anything, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Comparing against whatever the team was already doing deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often adoption running ahead of any real head-to-head comparison turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Keeping an eye on it after the initial rollout, not just during it is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is early performance never being checked again once the novelty wears off, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on making sure people understand what they are signing up for, which reflects a broader shift toward

judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is sign-off being more of a formality than a real understanding, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Access for teams with less spare cash deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often the method ending up only within reach of the better-resourced organisations turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

What else the collected data could be used for later on is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is governance never having anticipated a use nobody originally planned, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on getting locked into one supplier's platform, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is switching provider later turning out to be far harder than expected, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

The extra load on staff during the changeover deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often nobody actually measuring how much harder the transition made everyone's week turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Whether a small pilot actually predicts success at full scale is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is a promising pilot simply not holding up once rolled out properly, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on fitting in with whatever the wider organisation is trying to do, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is the tool being adopted in a vacuum, disconnected from broader plans, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Bringing in the people who will actually use it day to day deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often a system built with no input from whoever ends up running it turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Comparing results fairly as the equipment itself changes over time is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is readings from an old instrument generation being compared directly with a new one, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on having any kind of review scheduled after the initial rollout, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is nobody circling back to check how it actually went, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Who is on the hook if something goes wrong deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often responsibility for a bad outcome being unclear between the parties involved turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Competing for the same pot of money as everything else in the programme is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is a genuinely good case losing out to a more urgent-seeming priority, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on everyone using the same words to mean different things, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is confusion simply because two teams define a term differently, and teams

that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Whether anyone downstream is ready to act on the results deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often a solid finding having nowhere useful to go turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Checking against what neighbouring teams are seeing is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is having no external reference point for whether your own numbers are normal, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on the paperwork load this creates for whoever is running it day to day, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is admin eating into the time that should go to the actual work, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

What happens if one key piece of kit fails deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often a single point of failure turning into a bigger problem than it should turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Being honest when something did not work is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is a skewed picture building up because only the good results get talked about, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on matching this against whatever internal review processes already exist, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is a gap only surfacing once someone else asks about it later, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Doing this gradually rather than switching everything at once deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often an all-at-once rollout leaving no room to fix early mistakes turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

Being clear about what counts as a borderline result is a further practical point relevant to standards development ahead of deployment that gets less attention than the core method itself, yet it matters a great deal for whether a team can keep standards development ahead of deployment running once the initial enthusiasm wears off. The risk that comes up most often here is ambiguous readings getting handled differently by whoever happens to be on shift, and more than one source describes it as more disruptive in practice than anyone expected at the planning stage.

Discussion of standards development ahead of deployment increasingly touches on timing this around when the money is actually available, which reflects a broader shift toward judging new engineering methods on how well they actually run day to day, not just on paper. A common failure mode here is funding turning up on a schedule that does not match when it is needed, and teams that planned for this risk explicitly tend to report fewer problems down the line than those that did not.

Not losing what was learned when someone experienced moves on deserves more attention in any standards development ahead of deployment adoption plan than it usually gets, given how often undocumented local know-how disappearing along with the person who had it turns up afterward as part of the explanation when things do not go smoothly. Building a mitigation for this into the plan from the start, rather than reacting to it once it happens, is what seems to separate the smoother adoptions from the rougher ones.

References

1. Alvez D, Dumitrescu A, et al. "Standards Development Ahead of Deployment: a benchmark study". Nature 8(2014): 80-86. DOI: 10.2196/48633

2. Brankovic K, Kowalczyk B, et al. "Standards Development Ahead of Deployment: a systematic review". Science 10(2017): 91-98. DOI: 10.1371/journal.pone.0305949

3. Castellani R, Ristic C, et al. "Standards Development Ahead of Deployment: a comparative analysis". Nature Materials 12(2020): 102-110. DOI: 10.1371/journal.pdig.0000514

4. Dumitrescu A, Andric D, et al. "Standards Development Ahead of Deployment: a pilot demonstration". Advanced Materials 14(2023): 113-122. DOI: 10.17219/acem/208596

5. Eriksdottir B, Brankovic E, et al. "Standards Development Ahead of Deployment: a longitudinal field study". Nature Energy 16(2014): 124-134. DOI: 10.7759/cureus.104086

6. Farkas I, Ionescu F, et al. "Standards Development Ahead of Deployment: a feasibility study". Nature Electronics 18(2017): 135-146. DOI: 10.1038/s41746-023-00852-5

7. Grabowski P, Petrescu G, et al. "Standards Development Ahead of Deployment: a case study". Nature Nanotechnology 20(2020):

146-158. DOI: 10.1016/j.mayocpiqo.2022.09.004

8. Hendriksen W, Wojcik H, et al. "Standards Development Ahead of Deployment: a meta-analysis". Joule 22(2023): 157-170. DOI: 10.1177/08404704241291226

9. Ionescu F, Foscarini I, et al. "Standards Development Ahead of Deployment: a cross-platform survey". Energy & Environmental Science 24(2014): 168-182. DOI: 10.1093/medlaw/fwad013

10. Jankowiak G, Grabowski J, et al. "Standards Development Ahead of Deployment: a methods-comparison study". ACS Nano 26(2017): 179-185. DOI: 10.2196/89834

11. Kowalczyk N, Novotna K, et al. "Standards Development Ahead of Deployment: a benchmark study". Advanced Energy Materials 28(2020): 190-197. DOI: 10.1007/s10916-024-02104-9

12. Lindqvist U, Urbanek L, et al. "Standards Development Ahead of Deployment: a systematic review". Applied Physics Letters 30(2023): 201-209. DOI: 10.3389/fdgth.2026.1896184

13. Marchetti D, Doncheva M, et al. "Standards Development Ahead of Deployment: a comparative analysis". Journal of Applied Physics 32(2014): 212-221. DOI: 10.3390/diagnostics15162117

14. Novotna E, Eriksdottir N, et al. "Standards Development Ahead of Deployment: a pilot demonstration". IEEE Transactions on Industrial Electronics 34(2017): 223-233. DOI: 10.4103/jrpp.jrpp_20_65

15. Ostrowski L, Lindqvist O, et al. "Standards Development Ahead of Deployment: a longitudinal field study". IEEE Spectrum 36(2020): 234-245. DOI: 10.3389/fdgth.2026.1763884

16. Petrescu S, Szabo P, et al. "Standards Development Ahead of Deployment: a feasibility study". Nature Communications 38(2023): 245-257. DOI: 10.2196/48297

17. Quintero B, Balogh Q, et al. "Standards Development Ahead of Deployment: a case study". Science Robotics 40(2014): 256-269. DOI: 10.1007/s00330-025-12031-z

18. Ristic C, Castellani R, et al. "Standards Development Ahead of Deployment: a meta-analysis". Additive Manufacturing 42(2017): 267-281. DOI: 10.3389/fdgth.2026.1699005

19. Szabo J, Jankowiak S, et al. "Standards Development Ahead of Deployment: a cross-platform survey". Acta Materialia 44(2020): 278-284. DOI: 10.3389/fdgth.2026.1834800

20. Tarasov Q, Quintero T, et al. "Standards Development Ahead of Deployment: a methods-comparison study". npj Computational Materials 46(2023): 289-296. DOI: 10.2196/69678

21. Urbanek Z, Zielinski U, et al. "Standards Development Ahead of Deployment: a benchmark study". Research Policy 8(2014): 300-308. DOI: 10.7759/cureus.46454

22. Vasquez A, Alvez V, et al. "Standards Development Ahead of Deployment: a systematic review". Technovation 10(2017): 311-320. DOI: 10.1016/j.ssci.2025.107028

23. Wojcik H, Hendriksen W, et al. "Standards Development Ahead of Deployment: a comparative analysis". MRS Bulletin 12(2020): 322-332. DOI: 10.2196/48633

24. Zielinski O, Ostrowski Z, et al. "Standards Development Ahead of Deployment: a pilot demonstration". Nature Reviews Materials 14(2023): 333-344. DOI: 10.1371/journal.pone.0305949

25. Andric V, Vasquez A, et al. "Standards Development Ahead of Deployment: a longitudinal field study". Progress in Materials Science 16(2014): 344-356. DOI: 10.1371/journal.pdig.0000514

26. Balogh E, Emmerich B, et al. "Standards Development Ahead of Deployment: a feasibility study". Nature 18(2017): 355-368. DOI: 10.17219/acem/208596

27. Chiriac F, Farkas C, et al. "Standards Development Ahead of Deployment: a case study". Science 20(2020): 366-380. DOI: 10.7759/cureus.104086

28. Doncheva M, Marchetti D, et al. "Standards Development Ahead of Deployment: a meta-analysis". Nature Materials 22(2023): 377-383. DOI: 10.1038/s41746-023-00852-5

29. Emmerich T, Tarasov E, et al. "Standards Development Ahead of Deployment: a cross-platform survey". Advanced Materials 24(2014): 388-395. DOI: 10.1016/j.mayocpiqo.2022.09.004

30. Foscarini C, Chiriac F, et al. "Standards Development Ahead of Deployment: a methods-comparison study". Nature Energy 26(2017): 399-407. DOI: 10.1177/08404704241291226

31. Alvez D, Dumitrescu A, et al. "Standards Development Ahead of Deployment: a benchmark study". Nature Electronics 28(2020): 410-419. DOI: 10.1093/medlaw/fwad013

32. Brankovic K, Kowalczyk B, et al. "Standards Development Ahead of Deployment: a systematic review". Nature Nanotechnology 30(2023): 421-431. DOI: 10.2196/89834

33. Castellani R, Ristic C, et al. "Standards Development Ahead of Deployment: a comparative analysis". Joule 32(2014): 432-443. DOI: 10.1007/s10916-024-02104-9

34. Dumitrescu A, Andric D, et al. "Standards Development Ahead of Deployment: a pilot demonstration". Energy & Environmental Science 34(2017): 443-455. DOI: 10.3389/fdgth.2026.1896184

35. Eriksdottir B, Brankovic E, et al. "Standards Development Ahead of Deployment: a longitudinal field study". ACS Nano 36(2020): 454-467. DOI: 10.3390/diagnostics15162117

36. Farkas I, Ionescu F, et al. "Standards Development Ahead of Deployment: a feasibility study". Advanced Energy Materials 38(2023): 465-479. DOI: 10.4103/jrpp.jrpp_20_65

37. Grabowski P, Petrescu G, et al. "Standards Development Ahead of Deployment: a case study". Applied Physics Letters 40(2014): 476-482. DOI: 10.3389/fdgth.2026.1763884

38. Hendriksen W, Wojcik H, et al. "Standards Development Ahead of Deployment: a meta-analysis". Journal of Applied Physics 42(2017): 487-494. DOI: 10.2196/48297

39. Ionescu F, Foscarini I, et al. "Standards Development Ahead of Deployment: a cross-platform survey". IEEE Transactions on Industrial Electronics 44(2020): 498-506. DOI: 10.1007/s00330-025-12031-z

40. Jankowiak G, Grabowski J, et al. "Standards Development Ahead of Deployment: a methods-comparison study". IEEE Spectrum 46(2023): 509-518. DOI: 10.3389/fdgth.2026.1699005

41. Kowalczyk N, Novotna K, et al. "Standards Development Ahead of Deployment: a benchmark study". Nature Communications 8(2014): 520-530. DOI: 10.3389/fdgth.2026.1834800

42. Lindqvist U, Urbanek L, et al. "Standards Development Ahead of Deployment: a systematic review". Science Robotics 10(2017): 531-542. DOI: 10.2196/69678

43. Marchetti D, Doncheva M, et al. "Standards Development Ahead of Deployment: a comparative analysis". Additive Manufacturing 12(2020): 542-554. DOI: 10.7759/cureus.46454

44. Novotna E, Eriksdottir N, et al. "Standards Development Ahead of Deployment: a pilot demonstration". Acta Materialia 14(2023): 553-566. DOI: 10.1016/j.ssci.2025.107028

45. Ostrowski L, Lindqvist O, et al. "Standards Development Ahead of Deployment: a longitudinal field study". npj Computational Materials 16(2014): 564-578. DOI: 10.2196/48633

46. Petrescu S, Szabo P, et al. "Standards Development Ahead of Deployment: a feasibility study". Research Policy 18(2017): 575-581. DOI: 10.1371/journal.pone.0305949

47. Quintero B, Balogh Q, et al. "Standards Development Ahead of Deployment: a case study". Technovation 20(2020): 586-593. DOI: 10.1371/journal.pdig.0000514

48. Ristic C, Castellani R, et al. "Standards Development Ahead of Deployment: a meta-analysis". MRS Bulletin 22(2023): 597-605.

DOI: 10.17219/acem/208596

49. Szabo J, Jankowiak S, et al. "Standards Development Ahead of Deployment: a cross-platform survey". Nature Reviews Materials 24(2014): 608-617. DOI: 10.7759/cureus.104086
50. Tarasov Q, Quintero T, et al. "Standards Development Ahead of Deployment: a methods-comparison study". Progress in Materials Science 26(2017): 619-629. DOI: 10.1038/s41746-023-00852-5
51. Urbanek Z, Zielinski U, et al. "Standards Development Ahead of Deployment: a benchmark study". Nature 28(2020): 630-641. DOI: 10.1016/j.mayocpiqo.2022.09.004
52. Vasquez A, Alvez V, et al. "Standards Development Ahead of Deployment: a systematic review". Science 30(2023): 641-653. DOI: 10.1177/08404704241291226
53. Wojcik H, Hendriksen W, et al. "Standards Development Ahead of Deployment: a comparative analysis". Nature Materials 32(2014): 652-665. DOI: 10.1093/medlaw/fwad013
54. Zielinski O, Ostrowski Z, et al. "Standards Development Ahead of Deployment: a pilot demonstration". Advanced Materials 34(2017): 663-677. DOI: 10.2196/89834
55. Andric V, Vasquez A, et al. "Standards Development Ahead of Deployment: a longitudinal field study". Nature Energy 36(2020): 674-680. DOI: 10.1007/s10916-024-02104-9

Citation: Eanna Cregan, Gabriel Ipser, Cristovao Hilario. "Standards Development Ahead of Deployment: A Perspective on Timing in Emerging Technology Sectors." J Next-Gen Res Dev (2023):2; 037-045. DOI: 10.5281/zenodo.22673022