Clinical Decision Execution

A Missing Layer in Healthcare Computing

Clinical Decision Execution is the regulated layer that transforms executable clinical policy and longitudinal patient data into safe, auditable clinical action.

Mikko Tiihonen, MD
Forsante Ltd. • Helsinki, Finland • mikko.tiihonen@forsante.com
Version 1.0 • Published 23 November 2026 • Concept paper

ABSTRACT

Over more than a decade, Forsante has developed three systems for different clinical domains: Forsante Anticoagulation, Forsante PSA and IRMA. Despite their different purposes, these systems share an architectural pattern that is not adequately captured by the traditional concept of clinical decision support. Each transforms longitudinal patient data and executable clinical policy into patient-specific decisions, bounded actions and closed-loop follow-up. This paper examines that common architecture and proposes ***Clinical Decision Execution*** *as a distinct category of healthcare computing.*

Keywords: clinical decision execution · clinical decision support · executable clinical policy · clinical automation · closed-loop care · medical device software

From decision support to decision execution

Medicine excels at focusing on the disease. The strides that it has taken during the past century are impressive. These advances have enabled the development of effective therapies and detailed clinical guidelines for treating patients.

For many common chronic diseases, much of routine management is already described by these evidence-based guidelines. The remaining challenge is often not producing another recommendation, but reliably executing existing recommendations across large patient populations.

Even for one disease we may have over one hundred rules and dozens of facts that need to be kept in mind and evaluated. It has been estimated that following all the guidelines for a primary care physician's patients would require 26.7 hours of work per day (1).

Lack of knowledge is not healthcare's big problem. The burning issue is that despite all the knowledge, we are unable to implement it in care.

Clinical decision support has been offered as solution. It addresses uncertainty at the point of decision. Sometimes that is just what is needed. Yet, most of the time a physician meets ordinary people with ordinary ailments — long-established conditions requiring regular uneventful checkups and small adjustments to treatment. These decisions are mostly very straightforward and specified by the respective guidelines. The largely unaddressed problem is how to execute these decisions reliably at scale.

There is clearly a need for a new type of solution — **clinical decision execution (CDE)**.

Automation does not remove clinical governance. Healthcare has a long history of delegating predefined actions within constrained operating conditions. Protocol-driven nursing, PRN medication orders and algorithmic anticoagulation management are familiar examples. CDE extends this principle into software.

The promise of clinical decision execution is strong. If we could automate the obvious decisions, the ones that unequivocally follow from the guidelines and which do not endanger patient safety, then we could relieve doctors and nurses from that routine work – and remove one of the most expensive bottlenecks in care provision: the constraint of needing more healthcare personnel.

We will now examine more closely what clinical decision execution is and how it could transform the way we deliver care at scale.

How clinical decision execution (CDE) has emerged

CDE has been around for a while. Not as a separate category, but as a collection of individual developments, all pointing in the same direction. The anti-clotting medicine warfarin has been managed by algorithms successfully for hundreds of thousands of patients in many countries. Cancer patients have been followed up by algorithms which automate the alerting decision.

These systems differ from conventional decision support in an important way: the software is not limited to presenting information or recommendations for human consideration. Within a predefined operating envelope, it instigates the clinical action itself.

The need to execute clinical decisions has arisen from tedious, time-consuming parts of the clinical process, such as warfarin medication dosing. The first computerised systems for warfarin dosing date back to the 1980's. But their use became more widespread only after the adoption of electronic health record systems.

As an illustrative case we will examine Forsante Anticoagulation (2). What started as a simple SMS service for patients by the Helsinki University Central Hospital central laboratory in 2010, was developed and expanded into a CDE service enabling automation of the warfarin management in clinically safe cases. An 80% reduction in the time needed to manage the patients was reported for anticoagulation nurses (3).

Similarly, Forsante PSA has automated the follow-up of prostate cancer patients, enabling 9 out of 10 evaluations to reach the patient without needing a professional's input (4).

The common principle here is that a clinical guideline specifies the actions indicated in a clinical situation. When this situation can reliably be identified from available data, an executable implementation of the guideline can generate the corresponding action. If the patient state lies within the validated automation envelope, that action may be executed automatically.

Systems of this kind have appeared independently in different healthcare settings, but they share a common architectural pattern. Reliable identification of the validated automation envelope[1] in the state space, a reasoning mechanism for determining the required action, and an actuation mechanism for initiating that action are all necessary.

The missing part has been the terminology to adequately classify these systems. The current CDSS terminology has not acknowledged their existence.

These are not decision support systems. They are decision execution systems.

The relation to other autonomic systems

Other safety-critical domains have developed explicit models for levels of autonomy. SAE J3016, for example, is a classification for autonomous vehicles, starting from assistive systems at level 1, and reaching up to level 5 with total autonomy.

For medical devices, a similar grading has been suggested (5), the highest level for surgical robots reported being at Level 3 (conditional autonomy). Another proposed grading for machine-learning based clinical systems is a three-level grading, from assistive to "human-

[1] the set of patient states and operating conditions within which autonomous execution of a specific clinical policy has been validated

in-the-loop" to autonomous decisions. From 49 devices only 14 were in the "autonomous decisions" category, but even there were none that acted autonomously (6).

The previously mentioned Forsante Anticoagulation and Forsante PSA can autonomously execute predefined actions within a validated automation envelope and transfer control to a professional, when outside that space.

The execution gap

Clinical decisions are often made in a high-stakes, high-assurance situation. For safe autonomy, high reliability is a *sine qua non*, a requirement that cannot be relaxed.

With probabilistic AI solutions lower levels of autonomy dominate the landscape, and generative AI -based clinical solutions are predominantly assistive (7).

This leaves a significant gap between knowledge and execution to be filled by human action. This gap also locks away significant value that cannot be released on the lower levels of autonomy.

The requirement of a human-in-the-loop is in many ways problematic. Assistive solutions such as decision support tools are mostly sold on the premise that they save significant human work. The supervisory function is easy to explain as a non-demanding cursory task, but if done properly, it is often more demanding and taxing than doing the operation unassisted in the first place. This produces a perverse incentive to skip diligent verification in order to realise the promised time savings, resulting in the well-known "automation bias" (8).

To realise the promises of AI in healthcare, the value of automatic execution needs to be unlocked, and that requires higher levels of autonomy.

From decision support to decision execution

The category of decision execution has already emerged in practice, even though it has not previously been recognised as a distinct category.

Decision support is by definition assistive and subordinated to a human actor. The professional remains responsible for evaluating the recommendation and determining whether and how to act on it. Decision execution changes this relationship because the software can itself directly initiate or influence clinical management.

Decision execution is categorically different. Because decision execution can directly influence or initiate clinical management, its regulatory implications may be materially greater than those of purely informational software. The applicable medical device risk

classification depends on intended purpose and the potential consequences of the resulting decisions.

Where decision support advises, decision execution acts. Where decision support involves the clinician in the task, decision execution can execute the routine task on behalf of the clinician. Decision execution does not just deliver knowledge, it operationalises it, enabling integration into workflows and initiation of actions. It enables delivering on the promise: returning the clinicians' time for reallocation into higher value tasks.

CDE does not necessarily imply full automation. Rather, it introduces an execution-capable architecture in which predefined clinical actions may be carried out automatically when the validated conditions for execution are met.

Decision Support	Decision Execution
Advises	Executes
Human decides	Human governs
Knowledge delivery	Knowledge operationalization
Optional use	Workflow-integrated
Recommendation	Action generation

Historical case studies

To understand how decision execution has evolved, we will have a look at the two previously mentioned and widely deployed clinical decision execution systems from Finland.

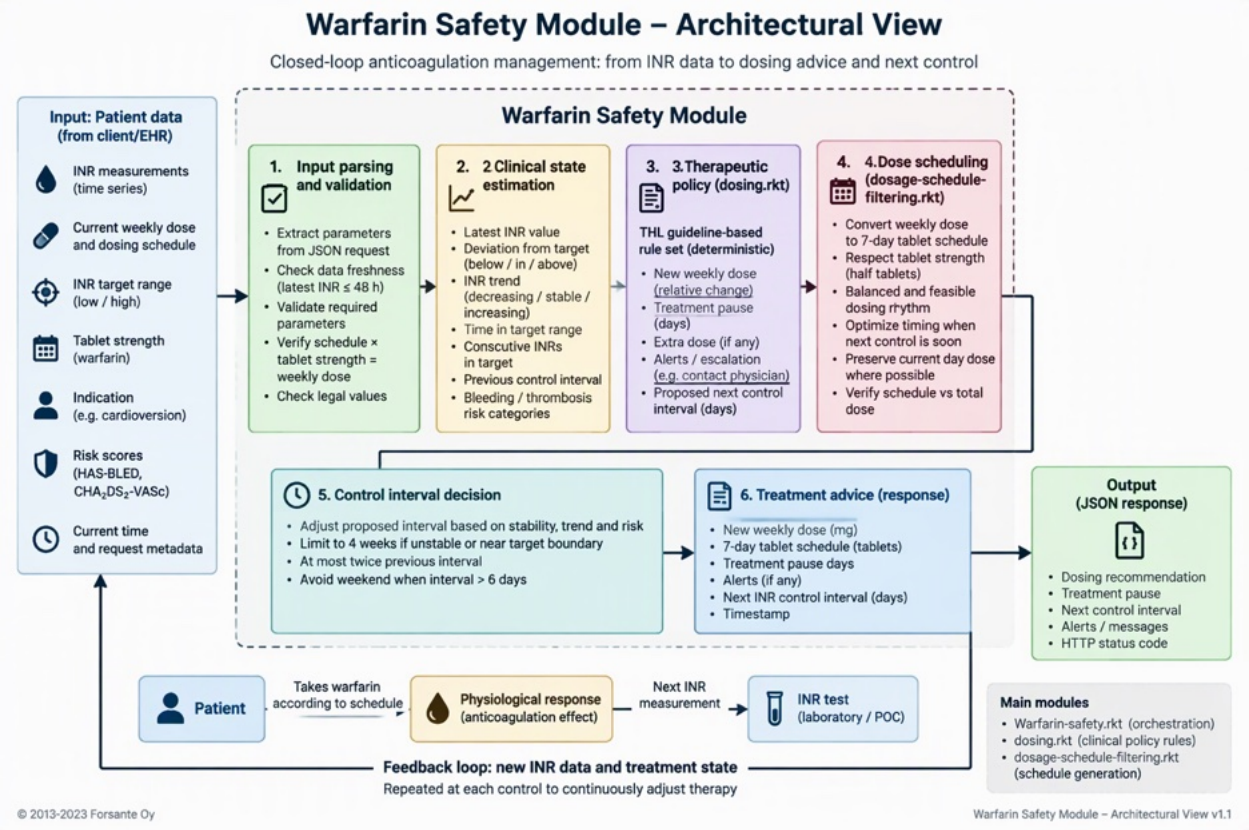
Forsante Anticoagulation

Forsante Anticoagulation operationalises the Finnish Institute for Health and Welfare (FIHW) guidelines for warfarin treatment management. It has been in clinical use since 2014. It implements a control loop, where the previous dosages and INR values drive the maintenance of treatment balance by adjustments of the weekly warfarin dosage, either automatically or under professional supervision, depending on the relation to the validated automation envelope. The rules in the FIHW guideline have been translated to program logic using pattern matching and decision tables.

The warfarin module in Forsante anticoagulation can be described as:

A deterministic therapeutic feedback controller that repeatedly observes the biological effect of treatment, estimates the patient's anticoagulation state, adjusts treatment, determines the intensity of subsequent observation, and escalates cases that leave the automated control region.

It is a concrete example of **executable clinical policy**: guideline knowledge isn't just displayed to the clinician. It is transformed into constrained rules that subsequently generate the patient-specific parameters for the next execution of the policy.



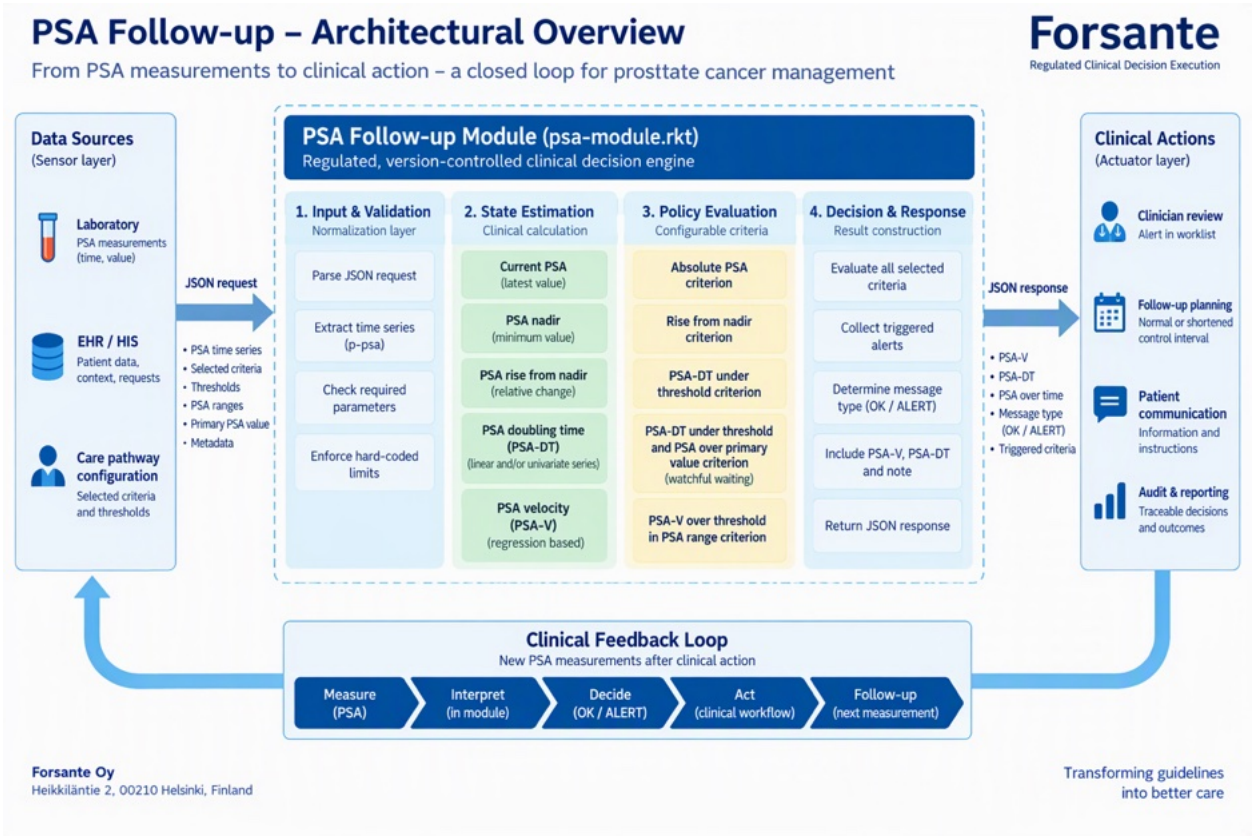
Forsante PSA

Forsante PSA enables automation of the post-treatment prostate cancer surveillance (4). It uses a deterministic time-series analysis engine that estimates the patient's PSA trajectory and applies explicit escalation rules. It serves as the decision component of a supervisory control loop for prostate cancer follow-up. It has a predefined set of escalation rules with adjustable limits, from which the clinician can build the patient-specific logic.

Forsante PSA supports both **event detection**, such as crossing an absolute or relative PSA threshold; and **trajectory detection**, such as a short doubling time or excessive PSA velocity.

The orchestration layer ensures that measurements occur on time; verifies that follow-up occurred; invokes the module with updated time-series when data arrive; converts results

into concrete clinical actions; assigns and tracks those actions; and determines the next follow-up interval.



The common architectural pattern

The common architecture is a reusable **clinical control architecture** in which disease-specific knowledge defines:

- how longitudinal observations are transformed into clinical state
- how that state is assessed within an executable clinical policy
- where autonomous execution is permitted
- what action follows
- and when the resulting state must be observed again.

The previous examples can be seen as two instances of the same architecture:

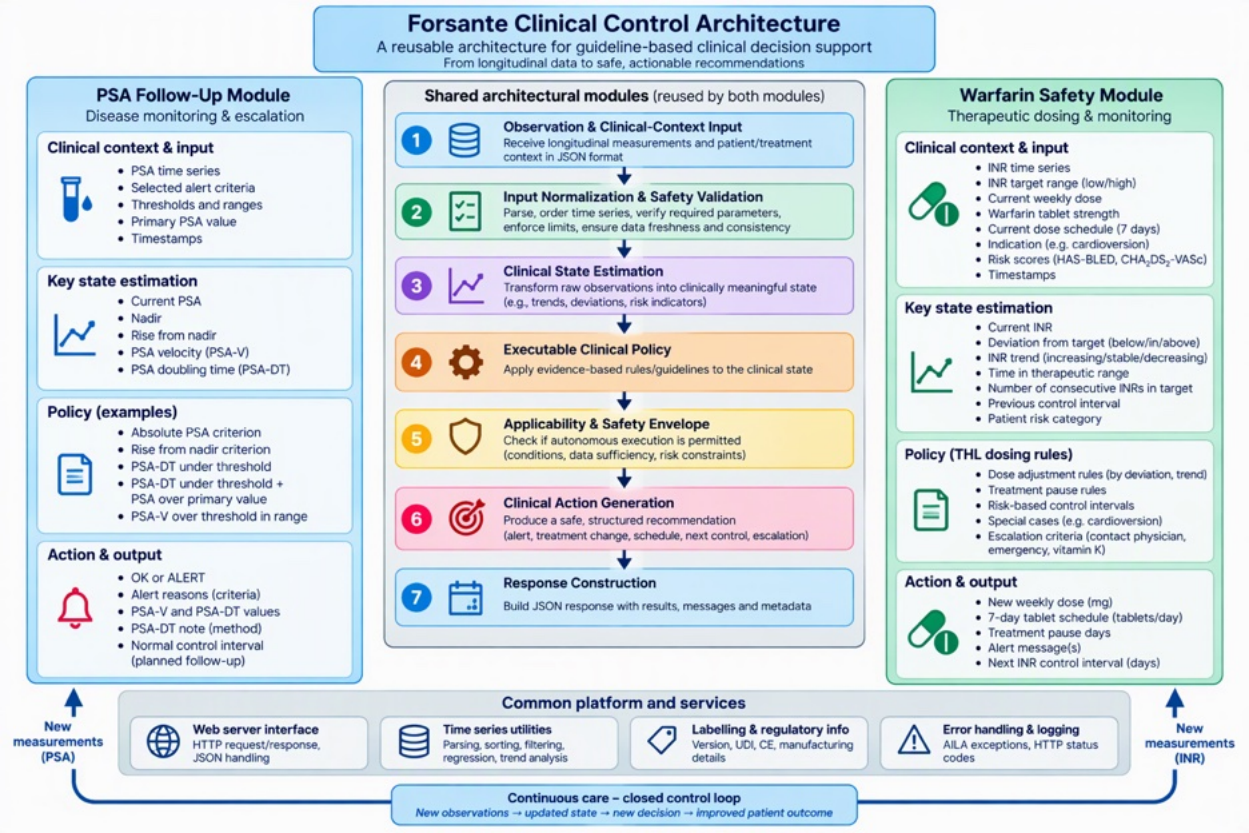
Forsante PSA: supervisory disease-progression controller

Forsante Anticoagulation: therapeutic + monitoring controller

And the reusable platform underneath can be thought of as:

$$Observation \rightarrow State \rightarrow Policy \rightarrow Safety\ Envelope \rightarrow Action \rightarrow Feedback$$

This provides an architectural foundation for **bounded autonomous execution of clinical guidelines**. A rules engine evaluates conditions. **The architecture executes a longitudinal clinical control policy**, including when to observe, when to act, and when to hand control back to a clinician.



Next evolutionary step

The next evolutionary step is to bridge the semantic gap between clinical guidelines and their operationalisation. A logic-programming approach based on modern non-monotonic reasoning with goal-directed execution is well-suited for this. It supports common-sense reasoning and allows expressing the guideline rules in a way that preserves the correspondence to the human-readable original text.

Forsante IRMA

Forsante IRMA has been designed to operationalise this approach in clinical practice. It provides a regulated clinical decision execution layer in which guideline-derived knowledge is expressed as modular rules and evaluated against the individual patient's clinical context. Goal-directed non-monotonic reasoning allows IRMA to deal with exceptions,

missing information and conflicting recommendations without requiring every possible clinical situation to be encoded as a decision tree. The resulting decisions are deterministic, explainable and traceable back to the input data, through the reasoning that produced them, enabling both professional decision support and automated clinical workflows.

Forsante IRMA interfaces with the surrounding technology stack using standard HL7 FHIR and CDS Hooks interfaces. This enables using it as a regulation-compliant deterministic clinical execution layer. Forsante IRMA is CE-marked as an MDR Class IIb medical device, providing a regulated foundation for clinical decision execution within its intended purpose.

Core architecture of Clinical Decision Execution

CDE does not replace healthcare systems of record or workflow systems. It forms a clinical control layer between longitudinal patient data and clinical action.

Architecture layer	Role	PSA example	Warfarin example
1. Clinical context & state	Transform longitudinal patient data into the clinical state needed by the policy	PSA level, nadir, trajectory, PSA-DT	INR, dose history, trend, stability
2. Executable clinical policy	Represent guideline or organisational policy as executable, testable logic	Surveillance and progression criteria	Dose and monitoring rules
3. Applicability & safety evaluation	Determine whether the patient lies within the **validated automation envelope**	Data sufficiency and defined kinetic criteria	Measurement validity, instability and extreme INR
4. Clinical decision	Determine what should happen next	Continue surveillance or escalate	Maintain/change/pause treatment
5. Action execution	Turn the decision into a concrete clinical or workflow action	Continue automatically or create clinical review	Generate dose schedule or physician contact
6. Closed-loop follow-up	Determine when to observe again and feed new observations into the next execution	Next PSA and reassessment	Next INR and reassessment
7. Governance & traceability	Ensure every execution is controlled, versioned, explainable and auditable	Trace alert to data, policy and algorithm version	Trace dose and interval to data, policy and algorithm version

Clinical control layer

The clinical control layer gives an answer to whether automation is safe in the patient's context:

"Can the action be executed safely, and what exactly should happen?"

Conceptually:

$$State \subset SafeAutomationEnvelope \rightarrow Automated\ execution$$

whereas:

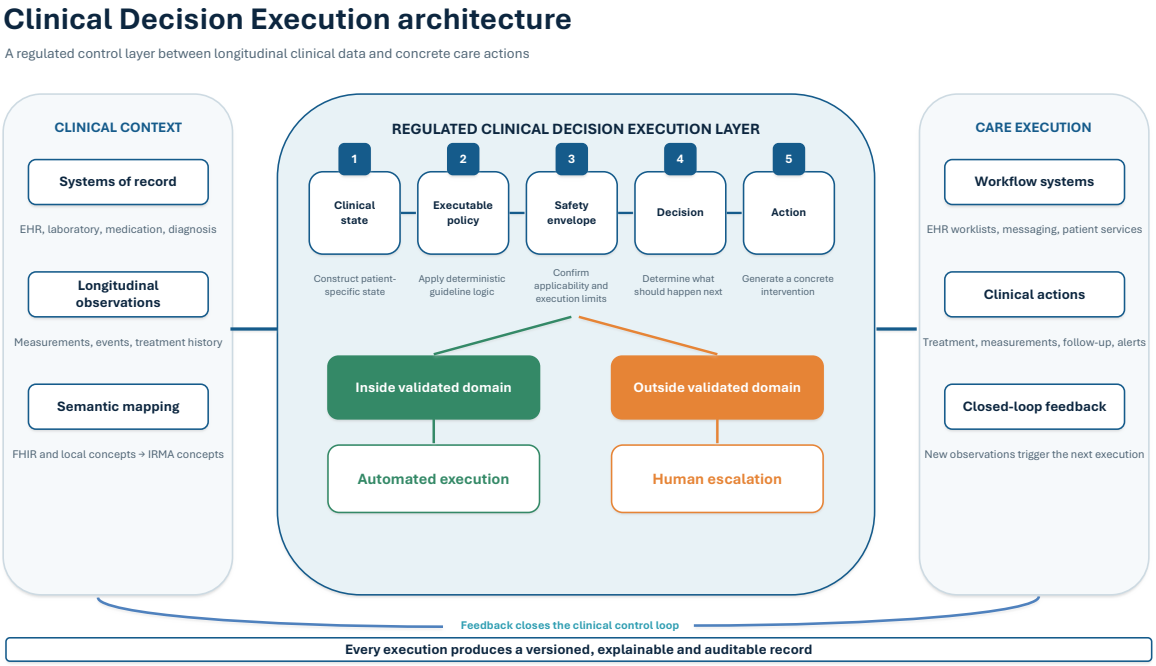
$$State \not\subset SafeAutomationEnvelope \rightarrow Human\ escalation$$

Importantly, it is not necessary to solve medicine in general.

It is only necessary to determine whether the current patient state lies inside a sufficiently constrained region for which the next action is defined and validated. The system executes a validated clinical policy only if the patient state lies within the policy's validated automation envelope.

Patient state	**CDE behavior**	**Human role**
Clearly within policy	Execute routine pathway	None, or oversight
Requires treatment adjustment but action is completely specified	Execute validated adjustment	Oversight according to intended use
Requires additional measurement	Schedule/request measurement	Usually none
Approaching safety boundary	Shorten follow-up or apply conservative action	Optional review
Outside validated automation envelope	Stop autonomous execution	Clinical assessment
Explicit high-risk state	Generate urgent escalation	Immediate professional action
Missing/inconsistent information	Do not make clinical inference	Correct data / clinician review

The generic architecture is in the figure below.



Clinical Decision Execution and multimorbidity

Patient pathways have been built for many diseases. The concept of a pathway is clear and compelling as far as the patient only has this one disease. Multimorbidity changes all that. Suddenly a clear pathway turns into a tangle of crossings, dead ends and general confusion.

The limitation is that the patient pathway is necessarily a simplified projection of clinical reality. It describes the trajectory from one guideline, whereas a real patient's care often deviates from that because of interacting diseases, treatments and varying clinical practices.

The examples presented here – solutions for warfarin treatment guidance and prostate cancer surveillance – are examples of single disease solutions. Their architecture has not been designed for interactions with other diseases, and they do not pretend to represent a clean pathway.

Clinical decision execution makes visible the conflicting constraints of the one-dimensional clinical patient pathways. This visibility enables the emergence of a new paradigm which allows for multimorbidity, guideline conflict resolution and does not artificially constrain the patient's clinical trajectory. The next logical extension of this concept is the multi-guideline patient management. Forsante has implemented the IRMA platform for this. Based on mathematical logic, it enables the concurrent execution of

multiple clinical guidelines and policies and resolution of the inevitable conflicts between them.

Future directions and implications

Healthcare has spent decades improving evidence-based clinical guidelines and deploying them into clinical practice. The next step is to make them executable and unlock the potential of automation within safe clinical constraints.

Clinical Decision Execution bridges the gap from guidelines to action, transforming validated guidelines and policies into deterministic and auditable actions within a validated automation envelope. Routine decisions for stable patients are a significant part of a clinician's workload. Being able to automate those decisions enables reallocating that effort to those tasks that need an expert's judgement.

The goal is not automated medicine. It is something more practical: autonomous execution of the routine clinical decisions that we already know how to make safely.

References

1. Porter J, Boyd C, Skandari MR, Laiteerapong N. Revisiting the Time Needed to Provide Adult Primary Care. J Gen Intern Med. 2023;38(1):147-55.
2. Tiihonen MT. Twelve-Year Real-World Evaluation of a Regulated Guideline-Based Warfarin Dosing and Care Automation System. medRxiv. 2026:2026.08.10.26360059.
3. Keränen T. Ohjelmistorobotti auttaa varfariinin annostelussa. Suomen Lääkärilehti. 2019;74(38):2074-5.
4. Bergroth R, Matikainen M, Rannikko A. Mobile PSA: A Novel Telehealth Tool for Prostate Cancer Follow-Up. Eur Urol Open Sci. 2021;28:43-6.
5. Lee A, Baker TS, Bederson JB, Rapoport BI. Levels of autonomy in FDA-cleared surgical robots: a systematic review. NPJ Digit Med. 2024;7(1):103.
6. Lyell D, Coiera E, Chen J, Shah P, Magrabi F. How machine learning is embedded to support clinician decision making: an analysis of FDA-approved medical devices. BMJ Health Care Inform. 2021;28(1).
7. Bakker M, Van Garderen A, Paget T, Zhao L, Naicker L, Radhakrishnan A, et al. Human in the loop in AI-enabled clinical decision support: a systematic scoping review and reporting checklist for lifecycle governance. Int J Med Inform. 2026;222:106701.
8. Goddard K, Roudsari A, Wyatt JC. Automation bias: a systematic review of frequency, effect mediators, and mitigators. J Am Med Inform Assoc. 2012;19(1):121-7.