# Monte Carlo Simulation and API 581

A high-level white paper on the business case for probabilistic simulation in integrity management — what it adds to API 581, and what it is worth to the operator.

Credosoft · 17 September 2026

*Written for integrity managers and the executives who sponsor them. It sets out the case in business terms and deliberately avoids calculation detail.*

## 1. Executive summary

Every integrity programme answers one recurring question: when does this equipment need to be inspected, repaired or replaced? API 581 answers it rigorously, and the answer it gives is a single number — a date, a remaining life, a risk band.

A number cannot say how confident anyone should be in it, how much is at stake if it is wrong, or which of its inputs actually drives it. Those are not technical footnotes. They are the questions a board asks when it approves an inspection budget, and the questions a regulator asks when a deferral is defended. Monte Carlo simulation exists to answer them. It does not change the physics or the standard: it runs the same API 581 calculation thousands of times, each time with a different but plausible set of inputs, and reports the range of outcomes and the probability of each.

The business case rests on two things. On safety, the risk that matters is the risk in the tail of the distribution — the unlikely outcomes that were never planned for — and simulation makes that exposure visible and quantifiable against a tolerance the company sets for itself. On cost, it moves inspection intervals in both directions rather than only shortening them, directs spending to where it buys the most risk reduction, and sizes turnaround scope at a confidence the business can budget against.

Monte Carlo itself is not an emerging technique. It has been a standard tool of engineering risk assessment for decades, and the question was never whether the method works. What has been missing in plant integrity is the framework to run it on: a standardised calculation engine and defensible damage models. Credo already provides that framework, in its API 581 engine and its damage modelling. The simulation layer therefore runs engineering the organisation already trusts — the same calculation that produces the compliant result, with uncertainty held as data that carries a stated basis rather than as an unexplained setting. It is built to be used as part of normal integrity work, not as a specialist exercise.

The deterministic result remains the compliance record. The simulation is the decision record.

## 2. Why a single number is not a decision

The question looks like scheduling. Behind it sits a decision under uncertainty: how much margin remains, how quickly it is being consumed, what it would cost to be wrong, and what would change the answer. API 581 is built precisely to organise that decision — damage mechanisms, damage factors, consequence, and a risk matrix that turns likelihood and consequence into a classification an organisation can act on.

The methodology is not the limitation. What the standard reports is a single value. A corrosion rate, a measured wall thickness and a minimum required thickness give a remaining life of 9.7 years. The arithmetic is correct and complete — but it is one outcome drawn from a range of plausible ones, presented as though it were the range itself. The rate depends on temperature, flow, composition and inhibitor performance. The thickness carries measurement scatter. The minimum required thickness may or may not include corrosion allowance.

The consequence is that every plan built on a point estimate is a plan at 50% confidence, and nobody has been told which 50%. Conservatism fills the gap, and that is correct engineering — but in a deterministic system conservatism is the only available way of expressing uncertainty, and its cost cannot be measured, argued or budgeted. A probabilistic treatment expresses the uncertainty directly and prices it.

This is an extension of the standard, not a departure from it. API 581 is already probabilistic in its foundations: inspection effectiveness enters the probability of failure through Bayesian updating, and the standard itself relies on simulation in places. What it reports, for practical and regulatory reasons, is a value and a classification. Monte Carlo changes the level at which that reasoning is reported, not the reasoning.

> **The governing rule**
> Monte Carlo does not replace API 581. The deterministic result remains the compliance record; the simulation is the decision record. The two must never disagree, which is why the simulation runs the same calculation rather than a second implementation of it.

## 3. The business case — safety

Safety is a question about the tail of a distribution, not its middle. Averages describe the typical future; incidents live in the futures that were judged unlikely and were not planned for. A method that reports only the middle has discarded the part of the distribution the safety question is about.

Simulation reports exposure directly: the probability that a target, a thickness limit or a risk threshold is crossed by a given date. An engineer can then say not only that the target date is 2031, but that the probability of crossing before it is 8%, rising to 30% by 2033. That is a statement about the risk the company is actually carrying, and it can be held against a tolerance the company has deliberately set — which is a governance decision, not a calculation.

It also shows how close an item sits to a change of classification. Risk matrices are step functions: two items a fraction of a percent apart in physics can fall in different bands, and the band is often what decides how frequently a plant is opened up. Simulation gives the probability of sitting in each band, which identifies the items where one more measurement would settle the decision — and the items where nothing is in doubt and effort is being spent for no return.

Two further points matter to a safety case. Likelihood and consequence call for different responses: high consequence with low likelihood is not fixed by inspecting more, and reporting the two only as a single product can hide that. And because every result carries its own uncertainty and the basis of its inputs, it can be defended to a regulator, an insurer or an internal audit.

> **The point**
> The mean serves the cost objective; the tail serves the safety constraint. Safety is not traded against cost — both are read from the same run.

## 4. The business case — operational efficiency

The safety case is the stronger one. The efficiency case is usually what pays for the work.

An inspection interval is a spending decision, and a probabilistic treatment moves it in both directions — extending it where the evidence supports the extension, shortening it where the tail says otherwise. Only the second direction is available to a conservative point estimate, which is why the deterministic answer is so often more expensive than it needs to be.

Spending can also be directed. Sensitivity analysis shows which inputs actually drive the answer. If the corrosion rate dominates, better thickness data sharpens the decision; if it does not, that money belongs somewhere else. Frequently the highest-value action turns out not to be an inspection at all, but the clarification of an unresolved data basis — which costs attention rather than budget.

It also re-ranks the work. Risk ranking says which item is worst. Budgeting asks where the next pound buys the most risk reduction, and those are not the same order. An inspection on a high-risk item that is already certain to be repaired changes no decision and buys nothing; one that resolves a genuine uncertainty can be worth several times its cost. Only a distribution supports that comparison, because the value of an inspection is the difference between two expected outcomes.

For turnarounds, the result reads as the probability that an item will need work in a given window: scope that can be sized at a stated confidence, long-lead material identified early, and fewer openings that resolve nothing. Across a register, every item is assessed by the same method and carries the same measure of confidence, so comparisons between units and between years become meaningful and the basis for a deferral is on the record rather than in the memory of the engineer who made it.

| The question | The deterministic answer | The probabilistic answer |
|---|---|---|
| **When must we act?** | A date | A date at a confidence we chose |
| **How much risk are we carrying?** | A band | The probability of each band |
| **What is driving it?** | Not available | A ranking of what actually matters |
| **What is an inspection worth?** | Not available | Expected risk reduction against its cost |
| **Where should the budget go?** | A risk ranking | A ranking by value, not by severity alone |

## 5. Why Monte Carlo

Simulation is sometimes presented as a new or experimental capability. It is neither. What has changed is not the method, but the availability of a framework strong enough to apply it to plant integrity — and that framework already exists in Credo.

- **It is a mature, proven method.** Monte Carlo simulation has been part of engineering risk management for decades, from probabilistic risk assessment in the nuclear industry to inspection planning for offshore structures and structural reliability analysis. It is recognised as a standard risk-assessment technique, and it has been used on high-consequence decisions for a generation. The question was never whether the method works.
- **What was missing was the framework, not the method.** Simulation needs something rigorous to run. It propagates uncertainty through a calculation, so without a standardised, auditable engine and defensible damage models it simply produces well-characterised answers to nothing in particular. Building and validating that engineering framework is the difficult and expensive part — not the sampling.
- **Credo already provides that framework.** Credo's API 581 engine and damage modelling already establish corrosion rates and damage factors for equipment in service, from the damage mechanisms the standard recognises. These are precisely the inputs a simulation needs, produced by the calculation the industry already trusts. That is what turns a proven technique into a practical one.
- **So the effort is a layer, not a programme.** It applies a mature method to engineering that already exists. There is no new physics to build, no new methodology to adopt, and no second implementation to validate. The deterministic result remains the compliance record throughout.

What simulation asks in return is discipline, and the discipline is what makes the result defensible. It needs inputs with a stated basis; where none exists, the honest answer is that the item cannot be assessed, not a number invented to fill the gap. It propagates uncertainty through a model without being able to correct one. And it is best applied by exception

— to the items where the decision is genuinely in play — rather than uniformly across a register.

## 6. How Credo approaches it

Credo is integrity management software for refineries and chemical plants. Credo8 provides the mature, data-centric core; Credo3D + AI adds 3D visualisation and analytical power on top of it. The API 581 engine and the damage-mechanism models are already part of that platform, establishing corrosion rates and damage factors for equipment in service. Probabilistic simulation is designed as a layer over that existing framework, not as a separate product and not as a new methodology.

That distinction is what de-risks the adoption. The engineering content of the assessment — the damage mechanisms, the standard's equations, the damage models — is already built. The simulation runs the same calculation the compliance result comes from, once per draw instead of once per assessment, so there is no second set of physics to maintain, no parallel methodology to reconcile, and no question about which answer is the compliant one. No organisation can afford two different answers to the same regulatory question, and by construction Credo produces only one.

Uncertainty, meanwhile, is treated as engineering data rather than as a setting. Every sampled input carries a distribution, a basis and a source, held as versioned configuration. Where no defensible basis exists, the item is reported as not assessable and the gap is named, rather than filled with an invented number. Each result carries its own margin of error and a record of what was varied and on what basis, so a probabilistic answer can be audited as readily as a deterministic one.

It is also built to be used rather than admired. Simulation is computationally heavy by nature, and the design target is that a register-sized assessment finishes in the time an engineer will wait, on hardware the company already owns, without disturbing the systems it reads from. The practical test of any simulation capability is whether it becomes part of routine integrity work or remains a specialist exercise; this one is aimed squarely at the first.

## 7. Conclusion — where Monte Carlo sits in the modern integrity landscape

API 581 remains the backbone, and nothing here suggests otherwise. It is the standardised, auditable methodology that turns damage mechanisms, consequence and inspection effectiveness into a risk classification, and it is what an integrity programme is written against. A probabilistic layer is not a competing methodology: it operates strictly inside it, on the same equations and the same inputs. That is not a matter of taste. A second, independently implemented probabilistic model would be an unmanageable risk in its own right, because it would eventually disagree with the standard calculation.

What has changed is the question being asked of integrity data. With an ageing asset base, constrained maintenance budgets and a continuing obligation to demonstrate that risk is understood, "what does the methodology require?" is no longer the only question. "How confident are we, what is driving the risk, and where does the next pound buy the most risk reduction?" is now asked of every integrity team, and a point estimate cannot answer it. This is the gap probabilistic simulation closes. The method itself is not new — it has been established for decades in nuclear, offshore and structural reliability practice — but it has only recently become practical in day-to-day integrity work, because it depends on a standardised calculation engine and defensible damage models to run on. That is the framework Credo already provides, and it is why the technique now belongs in the mainstream of plant integrity management rather than in the specialist's toolbox.

Comparatively, then: risk-based inspection tells an operator when to act. Monte Carlo tells them how confident they can be, what is driving the answer, what the alternatives would cost, and which items are one measurement away from a different decision. Used together — and only used together — they support the two things an integrity programme is ultimately judged on: that the plant is safe, and that the money spent keeping it safe was spent where it did the most good.

Credo's position is that these two should not be separate systems. The deterministic calculation and the simulation are the same engine, sampled or not, so the compliance record and the decision record can never drift apart.

> **In one line**
> API 581 answers the standard's question. Monte Carlo answers the operator's.

## 8. About Credosoft

Credosoft builds Credo — integrity management software for refineries and chemical plants. Credo8 provides the mature, data-centric core; Credo3D + AI adds 3D visualisation and AI-assisted analysis on top of it. Thirty years of global deployments help integrity teams move from fragmented legacy data to a single source of truth — and keep it there.