WHEN LIQUID OPERATIONS APPS BECOME THE OPERATIONAL RISK

HOW SCADA AND ENGINEERING LEADERS CAN STRENGTHEN AVEVA ENTERPRISE SCADA FOR SAFER, MORE DEFENSIBLE LIQUID PIPELINE OPERATIONS

A strategic guide for leaders responsible for throughput, compliance, and operator confidence in liquid pipeline operations

DEXCENT.COM | (833) 482-4100 | SALES@DEXCENT.COM

Executive Overview

LIQUID PIPELINE OPERATIONS DEPEND ON PRECISION.

Operators must understand what is moving, where it is, how it behaves in changing hydraulic conditions, and whether the information in front of them can be trusted. That trust is not created by SCADA screens alone. It is created by the configuration, integration, validation, and sustainment of the applications behind those screens.

For many pipeline organizations, Liquids Operations Apps inside AVEVA Enterprise SCADA are treated as technical functions within a larger SCADA modernization program. Metering, ticketing, tanking, batch tracking, pressure-flow logic, drag reducing agent logic, pipeline monitoring, reports, scheduling interfaces, and leak detection interfaces may each be discussed as separate work packages. Yet in daily operations, they do not behave as separate functions. **They form an operational system of systems.**

That distinction matters. When Liquids Apps are configured correctly, validated against real operating scenarios, and trusted by operators, they help the control room maintain flow, manage batches, improve accountability, support leak detection, and defend operational decisions. When they are only migrated, only connected, or only proven in narrow test conditions, they can create a different kind of risk. The system appears complete, but the operational logic may not be reliable under the conditions that matter most.

This is the risk many SCADA leaders are working to manage. They are not simply replacing a platform or upgrading applications. They are protecting the operating model of the pipeline. They are responsible for systems that must work across steady-state operations, transitions, abnormal events, maintenance windows, product movements, batch interfaces, changing densities, and the realities of human decision-making in the control room.

The challenge is that legacy applications often carry years of undocumented logic, workarounds, assumptions, and local knowledge. Mergers, acquisitions, rushed projects, and incremental upgrades can add more inconsistency. Over time, the organization may know that the system works most of the time, but have less certainty about why it works, where it is fragile, and what happens when conditions change.

A stronger approach begins by changing the question. Instead of asking, **"Are the Liquids Apps configured?"** senior SCADA and Engineering leaders should ask, **"Can we prove these applications support safe, compliant, and efficient operations under the conditions our pipeline actually experiences?"**

This eBook explores that question. It explains why Liquids Apps deserve executive and engineering attention, what **"right"** looks like in AVEVA Enterprise SCADA, why many projects struggle, and how Dexcent helps operators assess, modernize, validate, and sustain these critical applications. The goal is not to add complexity. It is to help leaders reduce uncertainty, strengthen operator trust, and move forward with a practical path that respects the operational reality of liquid pipelines.

CHAPTER 1

The Risk Inside Everyday Liquid Pipeline Operations

LIQUID PIPELINES ARE UNFORGIVING BECAUSE THEIR OPERATING CONDITIONS ARE DYNAMIC.

PRODUCTS CHANGE. DENSITIES CHANGE. BATCHES MOVE ACROSS LONG DISTANCES.

Interfaces develop between products. Tanks fill, drain, and switch service. Drag reducing agents affect flow behavior. Pressure and flow conditions can shift quickly. Scheduling, measurement, leak detection, and control room procedures all have to stay aligned.

In that environment, an application that appears to work in a normal state may still be fragile. The concern is not always a dramatic failure. More often, the concern is a gradual erosion of trust. Operators notice calculations that require interpretation. Engineering teams find exceptions that need manual review. Schedulers reconcile differences after the fact. Analysts maintain spreadsheets because the system does not give them the answer they need in the form they need it. These behaviours may look manageable, but they are early signs of operational risk.

MANY ORGANIZATIONS INHERIT THIS RISK HONESTLY.

A SCADA environment may have evolved over many years. Some Liquids Apps may have been implemented under time pressure. Others may have been migrated from a legacy system with limited documentation. Some logic may have been added to address a specific operational need at one site or one point in time. After mergers or acquisitions, different assets may bring different naming conventions, operating assumptions, and historical database structures.

The result is a system that works because people have learned how to work around its edges. That is different from a system that works because the applications are intentionally designed, configured, validated, and governed.

THE DIFFERENCE BECOMES VISIBLE DURING STRESS.

A batch transition, abnormal pressure event, communication issue, meter discrepancy, tank movement, or scheduling change can expose assumptions that were never fully tested. Operators may need to decide whether the data on the screen reflects the pipeline or whether the application is lagging, misaligned, or missing context. The more interpretation required, the more cognitive load moves from the system to the operator.

THIS MATTERS BECAUSE CONTROL ROOM DECISIONS ARE TIME-SENSITIVE.

Operators need systems that reduce ambiguity. When applications provide inconsistent, delayed, or poorly explained outputs, the control room becomes more dependent on memory, informal rules, and personal experience. Experienced operators may compensate effectively, but that creates a separate risk: the system becomes dependent on specific people rather than documented, repeatable logic.

THE SAME ISSUE AFFECTS ENGINEERING AND COMPLIANCE.

If a calculation cannot be explained, or if a report requires extensive manual reconstruction, the organization loses defensibility. It may still be able to justify a decision, but the effort required to do so increases. The cost appears later in audit preparation, incident review, root cause analysis, training, and project rework.

This is why Liquids Apps should not be treated as secondary applications within a platform program. They sit close to the operational decision. They influence how volume is accounted for, how batches are tracked, how interface losses are understood, how leak detection inputs are supported, and how operators build confidence in the information presented to them.

THE RISK IS RARELY THAT NOTHING WORKS.

The greater risk is that enough works to create confidence, but not enough has been validated to create certainty. Senior leaders do not need to assume failure. They need to know where the applications are dependable, where they are merely familiar, and where operating teams have been compensating for gaps the system should address.

A PRACTICAL MODERNIZATION EFFORT BEGINS THERE. IT DOES NOT BLAME THE PAST.

It recognizes that liquid pipeline operations have changed, platforms have changed, staffing models have changed, and expectations for reliability, compliance, and data integrity have increased. The question is **whether the application layer has evolved with those demands.**

CHAPTER 2

What Liquids Operations Apps Really Mean in Enterprise SCADA

THE PHRASE “LIQUIDS APPS” CAN SOUND NARROW, AS IF IT REFERS TO A SET OF

OPTIONAL MODULES OR CONFIGURATION TASKS.

In practice, Liquids Operations Apps are the applied operating logic that helps AVEVA Enterprise SCADA reflect the real behaviour of a liquid pipeline. They connect field measurement, operational context, business processes, and control room decisions.

THAT IS WHY THEY SHOULD BE UNDERSTOOD AS A SYSTEM OF SYSTEMS.

Metering and measurement integrity provide the foundation. If measured values are not reliable, timely, and properly contextualized, every downstream application is weakened. Ticketing and batch accounting depend on this foundation. Batch tracking requires alignment between product movement, linefill, schedule, interface handling, and operator visibility. Tanking functions must support inventory understanding, alarm context, and operational handoffs. Pressure-flow and hydraulic-related logic must make sense across changing operating modes, not only during stable conditions.

DRAG REDUCING AGENT LOGIC ADDS ANOTHER LAYER.

DRA can influence throughput and hydraulic behaviour, but its value depends on how well injection, monitoring, and operating assumptions are reflected in the system. If logic is poorly configured or difficult to interpret, operators may not have the confidence needed to act on it effectively.

INTERFACES MAKE THE ENVIRONMENT EVEN MORE INTERCONNECTED.

Scheduling systems provide essential context about planned product movements. Leak detection systems depend on accurate, timely, and consistent inputs. Operational reports depend on the quality of real-time, historical, and archived historical data. Custom scripts, action sequences, REFLEX logic, ACE logic, APL logic, and other custom programs may fill important operational roles, but they also create lifecycle responsibilities.

A PARTIAL IMPLEMENTATION CAN THEREFORE PRODUCE PARTIAL CONFIDENCE.

An interface can pass data while still failing to support the operational workflow. A calculation can be technically correct in one mode and misleading in another. A report can generate successfully while still requiring manual adjustment before it is useful. A migrated database can preserve values while losing context about how those values should be interpreted.

THIS IS WHERE MANY PROJECTS UNDERESTIMATE THE CHALLENGE.

Liquids Apps do not succeed because the software has capability. They succeed when the configuration reflects operating intent. That requires more than system knowledge. It requires understanding how the pipeline is run, how operators make decisions, how scheduling and leak detection teams use data, where exceptions occur, and what evidence the organization needs after the fact.

For example, batch tracking is not only a visualization problem. It is a trust problem. Operators need to understand where the batch is, what product is moving, what interface is expected, and when the data should trigger a decision. If the application does not match operational reality, operators will develop alternate methods. Once that happens, the official system and the practical workflow begin to diverge.

THE SAME IS TRUE FOR LEAK DETECTION INTERFACES.

The technical question is whether the interface exchanges data. The operational question is whether the data is fit for leak detection purposes.

- **Are values current?**
- **Are bad quality states handled correctly?**
- **Are communication interruptions understood?**
- **Are configuration changes controlled?**
- **Does the leak detection system receive the right information at the right time, with enough consistency to support confidence?**

AVEVA Enterprise SCADA can provide a strong platform for modern liquid pipeline operations, but platform capability is not the same as operational readiness. The application layer must be designed and validated around the pipeline's actual operating model. That includes real-time databases, historical databases, archived historical data, operational reports, external interfaces, custom logic, commissioning, and training.

THIS BROADER VIEW CHANGES HOW LEADERS SHOULD EVALUATE PROGRESS.

The milestone is not simply **"Liquids Apps installed."** A stronger milestone is **"Liquids Apps proven across the scenarios that define our operational risk."** That proof comes from design discipline, interface ownership, testing, commissioning, documentation, and hands-on knowledge transfer.

When Liquids Apps are treated as an operational system of systems, the project conversation changes. Teams stop asking only whether each component works in isolation. They begin asking whether the whole environment supports safe, efficient, auditable operations. That is the level at which these applications create value.

CHAPTER 3

The Operational Cost of "Good Enough" Liquids Apps

DEXCENT.COM | (833) 482-4100 | SALES@DEXCENT.COM

“GOOD ENOUGH” OFTEN SOUNDS REASONABLE IN TECHNICAL PROJECTS.

It suggests pragmatism, budget discipline, and avoidance of overengineering. In Liquids Operations Apps, however, good enough can become expensive because the costs are distributed across daily operations rather than concentrated in one obvious failure.

THE FIRST COST IS RECONCILIATION EFFORT.

When metering, ticketing, tanking, batch accounting, and historical data do not align cleanly, people spend time explaining variances. Some reconciliation is normal in liquid pipeline operations, but excessive manual reconciliation signals that the system is not carrying enough of the operational burden. Engineering, measurement, scheduling, and operations teams begin to spend time validating outputs that should already be trusted.

THE SECOND COST IS WEAKER LEAK DETECTION CONFIDENCE.

Leak detection depends on input quality, system alignment, and disciplined change control. If SCADA data quality is inconsistent, if interfaces are brittle, or if operating context is not properly represented, the leak detection program may still function, but confidence in its outputs can be reduced. That does not always mean an immediate compliance issue. It means the organization may have less certainty when certainty matters most.

THE THIRD COST IS OPERATOR DELAY.

Operators are trained to verify before acting. That discipline is essential. But when applications are mistrusted, verification can become a constant burden. An operator may pause to compare values across screens, call another team, check a spreadsheet, or rely on personal memory. In routine operations, that may only slow the workflow. In abnormal conditions, it can reduce the margin for error.

THE FOURTH COST IS AUDIT STRESS.

Audits and reviews expose weak documentation, unclear ownership, inconsistent configuration, and manual workarounds. The issue is not only whether the organization can produce evidence. It is how much effort is required to find it, explain it, and prove that the system reflects controlled operational intent. A well-configured and well-documented Liquids Apps environment makes defensibility easier. A poorly governed environment turns defensibility into a recovery exercise.

THE FIFTH COST IS DEPENDENCE ON TRIBAL KNOWLEDGE.

Every pipeline organization relies on experienced people. Their judgement is valuable. The risk appears when judgement becomes the glue holding the system together. If only a few people know why a calculation behaves a certain way, how a report should be corrected, or which interface values should be questioned, the organization has an availability problem disguised as expertise.

THESE COSTS CAN BE DIFFICULT TO QUANTIFY BECAUSE THEY ARE ABSORBED INTO NORMAL WORK.

- People stay late before audits.
- Operators create informal checks.
- Engineers answer recurring questions.
- Schedulers keep separate records.
- Analysts reconcile data.
- Project teams patch issues between releases.

None of these actions may appear severe on its own. Together, they indicate that the application layer is not providing the operational confidence it should.

CONTROL ROOM REALITY IS THE CLEAREST TEST.

Operators adapt quickly. If the system is unclear, they build patterns around it. If an alarm lacks context, they learn when to discount it. If a calculation is occasionally wrong, they learn when to verify it. If batch tracking is difficult to trust, they use another method. These adaptations are practical, but they create variation between operators, shifts, assets, and regions.

THAT VARIATION MATTERS FOR CONTROL ROOM MANAGEMENT DEFENSIBILITY.

The organization must be able to show that operators are supported by clear, reliable systems and consistent procedures. When applications require informal interpretation, the burden shifts to training, supervision, and after-the-fact explanation. In a complex liquid pipeline environment, that is not where leaders want risk to accumulate.

LIQUIDS APPS ISSUES RARELY ANNOUNCE THEMSELVES AS STRATEGIC RISKS.

They show up as small frictions: a report no one fully trusts, a batch calculation that requires a second check, a leak detection input that raises questions, an interface that works until a change occurs, or a commissioning test that proves connectivity but not operational performance.

The important insight is that Liquids Apps do not need to cause an event directly to increase operational exposure. They can reduce the quality of decision-making around the event. They can make it harder for operators to interpret what is happening. They can make it harder for engineering teams to explain what happened. They can make it harder for the organization to prove that the system behaved as intended.

THAT IS THE REAL COST OF GOOD ENOUGH.
IT DOES NOT SIMPLY REDUCE TECHNICAL ELEGANCE.
IT REDUCES OPERATIONAL CONFIDENCE.

CHAPTER 4

Liquids Apps in AVEVA Enterprise SCADA:

What “Right” Looks LikeChange Accelerates

A STRONG LIQUIDS OPERATIONS APPS ENVIRONMENT IS NOT DEFINED BY HOW MANY FUNCTIONS HAVE BEEN DEPLOYED.

It is defined by whether those functions support the pipeline's operating model with consistency, transparency, and control.

THE FIRST CHARACTERISTIC IS DETERMINISTIC BEHAVIOUR.

Operators and engineers need to know that calculations, logic, and interfaces behave predictably across operating modes. That includes steady-state operations, start-ups, shutdowns, batch changes, tank movements, communication interruptions, equipment changes, and abnormal conditions. The application should not be understood only by those who built it. It should be understandable to those who operate, maintain, and govern it.

THE SECOND CHARACTERISTIC IS TRANSPARENT LOGIC.

In many environments, custom logic accumulates over time. REFLEX, Action Sequence, ACE, APL, and other custom programs can be essential, but they need documentation, ownership, and lifecycle discipline. If logic is opaque, every future change carries more risk. If logic is transparent, teams can troubleshoot faster, train more effectively, and make better decisions during modernization.

THE THIRD CHARACTERISTIC IS ALIGNMENT ACROSS SCHEDULING, LEAK DETECTION, REAL-TIME DATA, HISTORICAL DATA, AND OPERATIONAL REPORTING.

These functions should not be treated as disconnected consumers of SCADA information. They are interdependent. Scheduling context influences how product movement is understood. Leak detection relies on timely and accurate data. Historical and archived historical databases support analysis, reporting, investigation, and audit evidence. Operational reports translate system data into decisions.

THE FOURTH CHARACTERISTIC IS CONTROLLED CONFIGURATION AND CHANGE MANAGEMENT.

Liquids Apps are not static. Pipelines change. Operating procedures change. Assets are added or removed. Interfaces evolve. Reports are adjusted. Logic is improved. Each change should be governed in a way that protects operations. That means understanding impact before implementation, validating changes against relevant scenarios, and preserving documentation after deployment.

THE FIFTH CHARACTERISTIC IS COMMISSIONING THAT REFLECTS REAL OPERATIONS.

Commissioning should prove more than installation. It should prove that applications support the workflows they are meant to support. For Liquids Apps, that can include validating metering, ticketing, tanking, batch tracking, pressure-flow logic, DRA logic, scheduling interfaces, leak detection interfaces, custom scripts, reports, and database behaviour in practical scenarios.

THIS IS WHERE AVEVA ENTERPRISE SCADA MODERNIZATION CAN CREATE LASTING VALUE.

The platform can support sophisticated pipeline operations, but the outcome depends heavily on design and configuration quality. Native capabilities matter. So do the decisions about how those capabilities are applied. A strong platform with weak operational definition will still create uncertainty. A well-defined operating model with disciplined configuration can improve trust, reduce rework, and strengthen sustainment.

Leaders should pay close attention to the boundary between software functionality and operating capability.

- Software functionality answers, **"Can the platform do this?"**
- Operating capability answers, **"Can our organization rely on this function during real operations?"**

The second question is more important.

Consider operational reports. A report that runs is functional. A report that gives the right stakeholders the right information, with trusted data, in a format they can use for decision-making or evidence, is operationally capable. The same distinction applies to interfaces. An interface that exchanges data is functional. An interface that preserves context, quality, timing, ownership, and downstream usability is operationally capable.

THE SAME PRINCIPLE APPLIES TO DATABASES.

Migrating real-time, historical, and archived historical data is not only a technical migration task. It is a continuity task. The organization needs confidence that data remains usable for trend analysis, reporting, investigation, and operational learning. If context is lost, the migration may be technically complete but operationally incomplete.

A STRONG LIQUIDS APPS ENVIRONMENT ALSO SUPPORTS PEOPLE.

Operators should understand what the applications are telling them and when to trust them. Engineering teams should understand how logic was designed and how to maintain it. Support teams should know ownership boundaries. Leaders should have confidence that documentation, validation, and training are sufficient to sustain the environment.

"WHAT RIGHT LOOKS LIKE" IS THEREFORE NOT PERFECTION.

It is disciplined, operationally grounded design. It is the ability to trace application behaviour back to operating intent. It is the ability to validate the system before go-live and sustain it after go-live. It is the ability to explain how data moves from field measurement to SCADA, from SCADA to scheduling and leak detection, from real-time systems to historical records, and from reports to decisions.

When this level of clarity exists, modernization becomes more than a platform project. It becomes an opportunity to strengthen the pipeline's operating model.

CHAPTER 5

Why Most Liquids App Projects Struggle

WHEN LIQUIDS APP PROJECTS STRUGGLE,

THE PROBLEM IS OFTEN BLAMED ON THE SOFTWARE, THE MIGRATION, OR THE COMPLEXITY OF THE ENVIRONMENT.

Those factors matter, but they are rarely the full story. **The deeper issue is usually the delivery model.**

Many projects begin with a technical scope and move quickly into implementation. Applications are installed or migrated. Interfaces are built. Databases are moved. Reports are recreated. Custom logic is translated. Testing confirms that functions execute. From a project management perspective, progress appears strong.

THE WEAKNESS APPEARS WHEN OPERATIONAL VALIDATION IS TOO NARROW.

A migrated application may match the legacy function without anyone asking whether the legacy function still reflects current operating needs. An interface may pass test messages without proving that the data supports scheduling or leak detection decisions. A report may reproduce old fields without confirming whether those fields still support audit, engineering, or control room requirements.

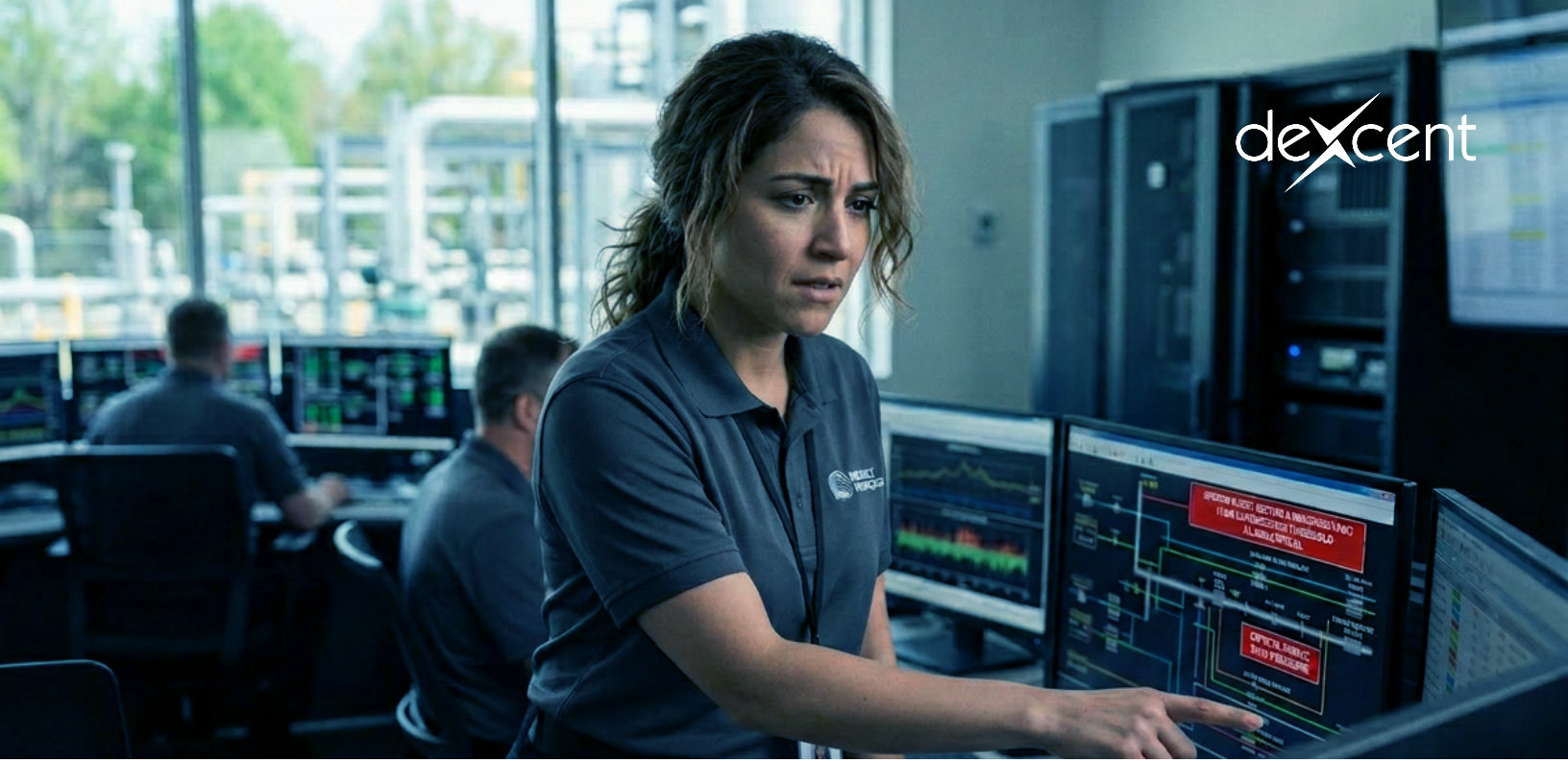
This pattern is common because SCADA projects are often under pressure. Cutover dates matter. Operations cannot be disrupted. Teams are balancing modernization with day-to-day responsibilities. The fastest path is to replicate what exists, fix obvious issues, and defer deeper questions. That can be reasonable in the moment, but it often preserves the weaknesses that modernization should have addressed.

ANOTHER FAILURE PATTERN IS SEPARATING LIQUIDS EXPERTISE FROM SCADA DELIVERY.

Liquids Apps require both platform knowledge and operational understanding. A team may understand AVEVA Enterprise SCADA but not the specific liquid pipeline behaviours that matter. Another team may understand operations but not the configuration and integration implications. If these perspectives are not brought together early, the project can produce technically correct outputs that do not fully serve the control room.

INTERFACES ARE ESPECIALLY VULNERABLE.

Scheduling and leak detection integrations cross organizational boundaries. They involve different owners, different data needs, and different support models. If no single team owns the end-to-end data flow, interface issues can become hard to diagnose. SCADA may send what it was asked to send. The external system may receive what it expected to receive. Yet the operational workflow may still be incomplete.



COMMISSIONING CAN ALSO BE TOO TECHNICAL.

Connectivity tests, point checks, and functional tests are necessary, but they are not sufficient. Liquids Apps need scenario-based validation. What happens during a batch transition? What happens when quality flags change? What happens when communications drop and recover? What happens when a schedule changes? What happens when a tank movement does not match expectation? What does the operator see, and what decision does the system support?

TRAINING IS ANOTHER COMMON GAP.

Too often, training is treated as documentation handover. Operators and engineers receive materials, attend sessions, and are expected to adapt. But application trust is built through practical use. Hands-on training, scenario walkthroughs, and clear explanation of logic help teams understand not only which buttons to press, but what the system means.

PROJECT CLOSE-OUT CAN CREATE ITS OWN PROBLEM.

Once the system is live, teams move on. Yet Liquids Apps require sustainment. Logic needs ownership. Reports need stewardship. Interfaces need monitoring. Database changes need governance. New operating conditions need evaluation. Without a sustainment model, the application layer begins to drift.

These struggles are avoidable. The delivery model has to start with operating intent, not only technical configuration. It has to define what the applications must do for operators, schedulers, leak detection teams, engineers, and compliance stakeholders. It has to document the desired behaviour before implementation. It has to test against real scenarios. It has to commission the full operating workflow. It has to train people in a way that builds trust.

THIS IS WHY A CONSULTING-LED APPROACH MATTERS.

Liquids Apps are not simply installed. They are translated from operational need into system behaviour. That translation is where risk is either reduced or carried forward.

CHAPTER 6

Dexcent's Approach to Liquids Operations Apps

DEXCENT.COM | (833) 482-4100 | SALES@DEXCENT.COM

DEXCENT APPROACHES LIQUIDS OPERATIONS APPS AS

OPERATIONAL ASSETS, NOT ISOLATED CONFIGURATION TASKS.

The goal is to help SCADA and Engineering leaders move from uncertainty to a controlled, validated application environment that supports the way the pipeline actually runs.

THAT BEGINS WITH DEFINITION.

Before configuration or migration decisions become fixed, the team needs a clear understanding of operating intent. What decisions must the applications support? What data must be trusted? Which systems depend on SCADA outputs? Which workflows cross scheduling, leak detection, operations, engineering, measurement, and reporting? Which legacy behaviours should be preserved, improved, or retired?

This definition step is important because it prevents the project from becoming a blind replication of the past. In many environments, legacy logic contains both valuable knowledge and accumulated workarounds. Dexcent's role is to help separate the two. Some existing behaviours should be protected because they reflect operating reality. Others should be questioned because they reflect old constraints, undocumented fixes, or outdated assumptions.

A SOLUTION DEFINITION DOCUMENT CAN PROVIDE THE STRUCTURE FOR THIS WORK.

It defines the functions and features required, captures the intended behaviour, clarifies ownership, and creates a reference point for implementation, testing, and rollout. For Liquids Apps, this type of definition is especially valuable because so many functions depend on shared data and cross-system logic.

From there, Dexcent supports the practical work of Liquids Apps configuration and fine-tuning. This includes metering, ticketing, tanking, batch tracking, pipeline monitoring, pressure-flow functions, DRA-related functionality, and other applications that support liquid pipeline operations. Configuration is treated as an operational design activity. The question is not only whether values display correctly. The question is whether the configured behaviour supports the decision the operator or engineer needs to make.

INTERFACES RECEIVE THE SAME DISCIPLINE.

Dexcent supports the migration, creation, and fine-tuning of interfaces to pipeline scheduling systems and leak detection systems. These interfaces are not just data paths. They are operating dependencies. The work must account for data timing, quality, context, exception handling, ownership, and lifecycle support. When interfaces are designed with operational use in mind, they reduce ambiguity between teams and improve confidence in downstream systems.

DATABASE MIGRATION IS ANOTHER CORE AREA.

Real-time databases, historical databases, and archived historical databases must be migrated and configured in ways that preserve operational continuity. This work is not only about moving data. It is about maintaining the usefulness of that data for analysis, reporting, investigation, audit readiness, and future improvement. A well-executed database migration helps teams trust both current operations and historical evidence.

Dexcent also supports the migration, creation, and sustainment of scripts, applications, REFLEX logic, Action Sequence, ACE, APL, and custom software related to SCADA and liquid operations. This work is often where undocumented risk lives. Custom logic may be essential, but it needs to be understood, reviewed, documented, and validated. The value is not simply in recreating custom behaviour. The value is in making that behaviour maintainable.

OPERATIONAL REPORTING IS ANOTHER IMPORTANT PIECE.

Reports should not be treated as project extras. They are often how leaders, engineers, compliance teams, and operational stakeholders understand performance and defend decisions. Dexcent supports the migration and creation of operational reports so that reporting aligns with trusted data, defined logic, and practical business needs.

COMMISSIONING BRINGS THESE PIECES TOGETHER.

Liquids Applications should be commissioned against the real workflows they support. That includes validating applications, interfaces, databases, custom logic, and reports as a connected operational environment. The objective is to reduce cutover risk and increase confidence before the system becomes the control room's daily reality.

HANDS-ON TRAINING COMPLETES THE PATH FROM TECHNICAL DELIVERY TO OPERATIONAL ADOPTION.

Operators and engineering teams need more than a walkthrough. They need to understand how the application behaves, what the logic means, how exceptions should be interpreted, and where ownership sits. Effective training helps prevent the return of informal workarounds after go-live.

DEXCENT'S BROADER STRATEGIZE, TRANSFORM, EVOLVE APPROACH FITS NATURALLY HERE.

IN THE STRATEGIZE PHASE, LEADERS CLARIFY THE CURRENT STATE, DEFINE THE DESIRED OPERATING MODEL, ASSESS APPLICATION GAPS, MAP DEPENDENCIES, AND PRIORITIZE THE WORK.

This may include a Liquids Operations Apps assessment, interface review, database review, reporting review, or readiness discussion ahead of a larger AVEVA Enterprise SCADA modernization program.

IN THE TRANSFORM PHASE, THE DEFINED WORK IS IMPLEMENTED.

Applications are configured or migrated. Interfaces are built or refined. Databases are migrated. Custom logic is created or rationalized. Reports are developed. Commissioning is performed. Teams are trained. The goal is controlled change that supports operational continuity.

IN THE EVOLVE PHASE, THE ENVIRONMENT IS SUSTAINED.

Applications need governance, performance review, change management, and periodic reassessment. As operations change, the application layer should not drift into a new set of undocumented assumptions. It should continue to mature with the pipeline.

This approach matters because it connects technical services to practical outcomes. Reduced cutover risk. Stronger operator trust from day one. Clearer interface ownership. Better audit readiness. More maintainable custom logic. More reliable reporting. Better continuity between real-time operations and historical evidence.

DEXCENT IS NOT THERE TO MAKE THE SYSTEM MORE COMPLEX.

The work is to help leaders simplify the path to operational confidence by bringing structure to the application layer. In liquid pipeline operations, that structure is what allows modernization to hold under pressure.

CHAPTER 7

Indicators You Are Ready for a Liquids Apps Assessment

A LIQUIDS APPS ASSESSMENT IS VALUABLE WHEN LEADERS NEED A CLEARER VIEW OF

APPLICATION RISK BEFORE A MAJOR PROJECT, AUDIT, EXPANSION, OR OPERATIONAL CHANGE.

It is also valuable when the organization senses friction but lacks a structured way to diagnose it.

One signal is recurring manual reconciliation after batches. If teams regularly need to explain differences between systems, recreate movements, or validate accounting outside the SCADA environment, the application layer may not be providing enough confidence. The issue may sit in measurement, ticketing, batch tracking, tanking, historical data, reporting, or the handoffs between them.

ANOTHER SIGNAL IS OPERATORS BYPASSING OR SECOND-GUESSING SCADA CALCULATIONS.

This does not mean operators are doing something wrong. It means they have learned that certain outputs require caution. That caution may be justified, but it should be understood. Leaders need to know which calculations are trusted, which are questioned, and why.

LEAK DETECTION CONFIDENCE ISSUES ARE ANOTHER IMPORTANT INDICATOR.

If leak detection teams frequently question SCADA inputs, data quality, timing, or interface behaviour, the organization should not treat the issue as a simple integration concern. It may reflect deeper problems in configuration, data governance, or change control.

UPCOMING AUDITS CAN ALSO JUSTIFY AN ASSESSMENT.

Waiting until audit preparation begins can force teams into evidence recovery mode. A better approach is to understand documentation, reporting, ownership, and validation gaps before the organization is under pressure to produce answers.

EXPANSIONS, ACQUISITIONS, AND MAJOR OPERATING CHANGES ARE ALSO STRONG TRIGGERS.

New assets, new products, new operating modes, or new organizational structures can expose inconsistencies across applications. If different assets follow different conventions or rely on different legacy assumptions, the gap may not be visible until integration becomes necessary.

AGING CUSTOM LOGIC IS ANOTHER SIGNAL.

If important scripts or programs are poorly documented, owned by a small number of people, or difficult to change safely, the organization is carrying sustainment risk. The question is not whether the logic works today. The question is whether the organization can maintain and defend it tomorrow.

REPORT DISTRUST IS ALSO REVEALING.

If leaders or engineers rely on reports but routinely ask for manual confirmation, the reporting layer is not doing its job. Reports should reduce uncertainty, not create another verification loop.

LEADING OPERATORS DO NOT WAIT FOR FAILURE BEFORE ASKING THESE QUESTIONS.

They assess before upgrading. They validate before commissioning. They train before go-live. They define ownership before issues cross team boundaries. They treat Liquids Apps as part of the operating model, not as a technical afterthought.

AN ASSESSMENT SHOULD NOT BE POSITIONED AS A FAULT-FINDING EXERCISE.

Its value is clarity. It helps teams identify where applications are strong, where documentation is weak, where interfaces need better ownership, where custom logic needs review, where operators have developed workarounds, and where modernization would create the most value.

For senior SCADA and Engineering leaders, this clarity can improve project planning. It can reduce scope uncertainty. It can help justify investment. It can surface risks early enough to manage them. It can also help internal teams align around a shared view of what needs to change.

The best time to assess Liquids Apps is before the organization is forced to explain them under pressure.

CHAPTER 8

From Assessment to Outcomes

DEXCENT.COM | (833) 482-4100 | SALES@DEXCENT.COM

THE VALUE OF IMPROVING LIQUIDS OPERATIONS APPS

IS NOT MEASURED BY CONFIGURATION COMPLETION ALONE. IT IS MEASURED BY BETTER OPERATIONAL OUTCOMES.

ONE OUTCOME IS IMPROVED BATCH ACCOUNTABILITY.

When batch tracking, metering, ticketing, tanking, and reporting are aligned, teams have a stronger basis for understanding product movement and resolving questions. This does not eliminate the complexity of liquid pipeline operations, but it reduces unnecessary ambiguity.

ANOTHER OUTCOME IS REDUCED OPERATOR COGNITIVE LOAD.

Operators should not have to carry the full burden of interpreting inconsistent data, checking multiple sources, or remembering informal exceptions. Strong applications provide clearer context, more predictable behaviour, and more usable information. That helps operators focus on decisions rather than detective work.

A THIRD OUTCOME IS STRONGER LEAK DETECTION PROGRAM INPUTS.

Leak detection performance depends partly on the quality and reliability of the information it receives. Improving SCADA data quality, interface behaviour, and change control can support greater confidence in the inputs that leak detection systems depend on.

A FOURTH OUTCOME IS FASTER INCIDENT RESPONSE AND REVIEW.

When applications are transparent, data is reliable, reports are useful, and historical information is preserved, teams can understand events more quickly. They can spend less time reconstructing what happened and more time responding, learning, and improving.

A FIFTH OUTCOME IS STRONGER AUDIT READINESS.

Audit confidence comes from more than documentation. It comes from the ability to show that applications were designed, configured, validated, commissioned, and sustained with operational intent. When that evidence is available, leaders are not forced to rely on memory or last-minute reconstruction.

THERE ARE ALSO IMPORTANT HUMAN OUTCOMES.

Operator trust improves when the system behaves predictably and training explains why. Engineering confidence improves when logic is documented and maintainable. Organizational ownership improves when interfaces, reports, databases, and custom applications have clear accountability. These outcomes may be less visible in a project schedule, but they are essential to long-term performance.

THE STRONGEST RESULT IS OPERATIONAL CONFIDENCE.

Leaders know what the system is intended to do. Operators understand how to use it. Engineers know how to support it. External systems receive information that is fit for purpose. Reports provide useful evidence. Changes can be made without rediscovering the entire environment.

THAT IS WHAT MODERNIZATION SHOULD DELIVER.

Not more screens. Not more complexity. A better operating foundation.

A CONVERSATION WORTH HAVING

If your Liquids Operations Apps have not been questioned recently, that should prompt a serious discussion.

Not because they are failing. Not because the team has done poor work. Because liquid pipeline operations change, SCADA platforms evolve, people move roles, assets are added, interfaces age, and operating assumptions drift. Applications that once fit the business can become misaligned with how the pipeline now runs.

For SCADA Directors, SCADA Managers, Engineering Managers, and technical leaders, the opportunity is to address that drift before it becomes operational friction, audit stress, or cutover risk. The right conversation can clarify where your Liquids Apps are strong, where they may be carrying inherited assumptions, and where modernization could produce practical value.

Dexcent helps pipeline operators have that conversation from an operator-first perspective. The focus is not on selling a generic upgrade. It is on understanding how your AVEVA Enterprise SCADA environment supports metering, ticketing, tanking, batch tracking, pressure-flow logic, DRA, scheduling interfaces, leak detection interfaces, databases, reports, custom logic, commissioning, and training.

A Liquids Operations Apps Readiness Discussion gives your team a structured way to examine the application layer before decisions become expensive. It can help you identify the right next step, whether that is an assessment, a modernization plan, interface review, commissioning support, training, or a broader Enterprise SCADA modernization roadmap.

The goal is simple: **help your organization operate with stronger confidence in the applications that support safe, compliant, and efficient liquid pipeline operations.**