CONVIVA | Intellyx™

# Optimizing Customer Experience with Real-Time Performance Analytics

## Intellyx Analyst Guide for Conviva

By Jason Bloomberg and Jason English, Intellyx

# Table of Contents




## Introduction

The future prospects of a business—and let's face it, just about every business is software-driven now, are often predicted by the gold standard metric of NPS, or net promoter score, which measures how likely a customer would be to recommend the company to a friend or colleague.

That's not going to cut it for today's digital enterprise. We need deep observability and contextual telemetry data for real-time insights about the quality of experience (QoE) encountered by individual customers as they work with our digital products and services.

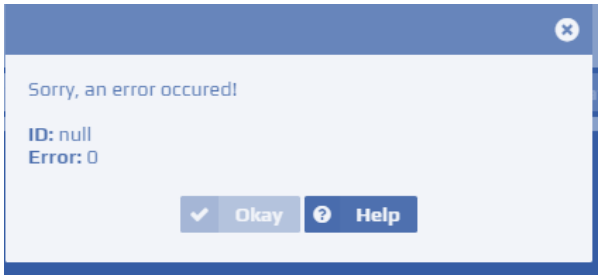
This 4-part Intellyx Analyst Guide, underwritten by **Conviva**, will take operators and architects through the observability and data transformations needed to optimize the customer journey.

**By Jason Bloomberg**

Managing Director & Analyst
Intellyx

# Avoiding Conversion Blind Spots by Focusing on 'Quality of Experience'

## When making a purchase online, have you run into any of the following problems?




- You try searching for a product, but the search results took too long.
- Your search results turned up zero items—even though you know there are available products that would meet your criteria.
- You select an item or even put it into your shopping cart, but then some kind of error pops up, and you have to start over,
- Instead of an error, you just get the dreaded spinning wheel—and you must still start over,
- You finally get to the end, but the site freezes when you submit payment. Did it go through or not?
- You complete the purchase, but then the confirmation doesn't arrive. How long should you wait?

As consumers, we all know such problems are maddeningly common.

Now, let's put on our operator's hat. Our job is to keep the site running smoothly—and quickly. We have a plethora of monitoring and analytics tools to help us.

Just one problem: *observability tools won't help us solve any of the problems above*. Even real-user monitoring (RUM) won't give us the information we need.

Something important is missing from our operator's tool belt. Where's the blind spot, and how do we fix it?

# Why Observability doesn't Solve the Quality of Experience Problem

Observability tools glean insights from system or server telemetry data – logs, metrics, and traces from various applications and systems that indicate their respective internal states at particular points in time.

Product analytics tools analyze how users engage with a product or service, giving product teams the ability to track, visualize, and analyze user engagement and behavior data.

Such telemetry or user engagement data provide some insight into end-users' experience to be sure – but the connection between such telemetry and the ***quality of experience*** **(QoE)** of users at the interface is at best indirect.

Fundamentally, observability is *system-centric* rather than *experience-centric*.

Product analytics tools that track user engagement data and other experience-centric tooling like RUM also fall short, because they lack the ability to show the full picture.

Such tools can show what happened in the past, but they don't *explain* those events – and they don't work in real-time regardless.

As a result, operators must 'go fishing' in the sea of system and server telemetry to uncover root causes, wasting time and resources – thus adversely impacting consumers.

Objective understanding and measurement of the quality of consumers' experience over time, in fact, are the missing elements for all the ecommerce examples above.

***It's not good enough to track whether some event took place – did the search return results? Did the user abandon the cart? Did the purchase complete? Did the purchase confirmation arrive?***

Operators must also track how much time various actions required to gain a true picture of the QoE – and hence, how well user-facing processes successfully complete.

## Struggling with Conversion Blind Spots

*Conversion* is the marketer's term for the most important landmark along any customer's journey: the moment they complete a purchase.

Converting prospects into customers, of course, is of utmost importance to any company looking to do business. From the end-user's perspective, however, it is but one part of the journey.

For such individuals, such journeys are *stateful* – a connected sequence of events that build upon and relate to one another.

This statefulness is where observability and product analytics tools fall short, as they focus on events in time, rather than how they connect to form a coherent experience for users.

Without a QoE platform like Conviva's, operators and marketers alike are left scratching their heads over why users are abandoning processes mid-stream – leading not only to lost revenue, but also complaints on social media, poor reviews, and other negative consequences.

The core problem is that existing observability tools and the teams that operate them are disconnected from user experience and engagement – qualities that matter more to the business than the performance and uptime metrics that operators traditionally focus on.

## The Need for Experience-Centric Operations

Performance and uptime are still important, of course – but in today's digital world, operators should focus more on higher-level user experience metrics.

Operators must monitor every user journey across every user in real-time. If users are having problems – with logging in, searching, making purchasing decisions, completing purchases, or any other aspect of their journey – then QoE tooling should immediately alert operators to the issue.

Experience-centric operations connects a comprehensive measurement of user experience to the underlying system and application performance metrics, enabling operators to focus on the data that matter to users – and hence, to the business.

Experience is more than a simple count of events or metrics. Understanding the complexities of the user journey requires complex calculations of timing, time intervals, sequences, and the statefulness of the user experience – in other words, stateful analytics.

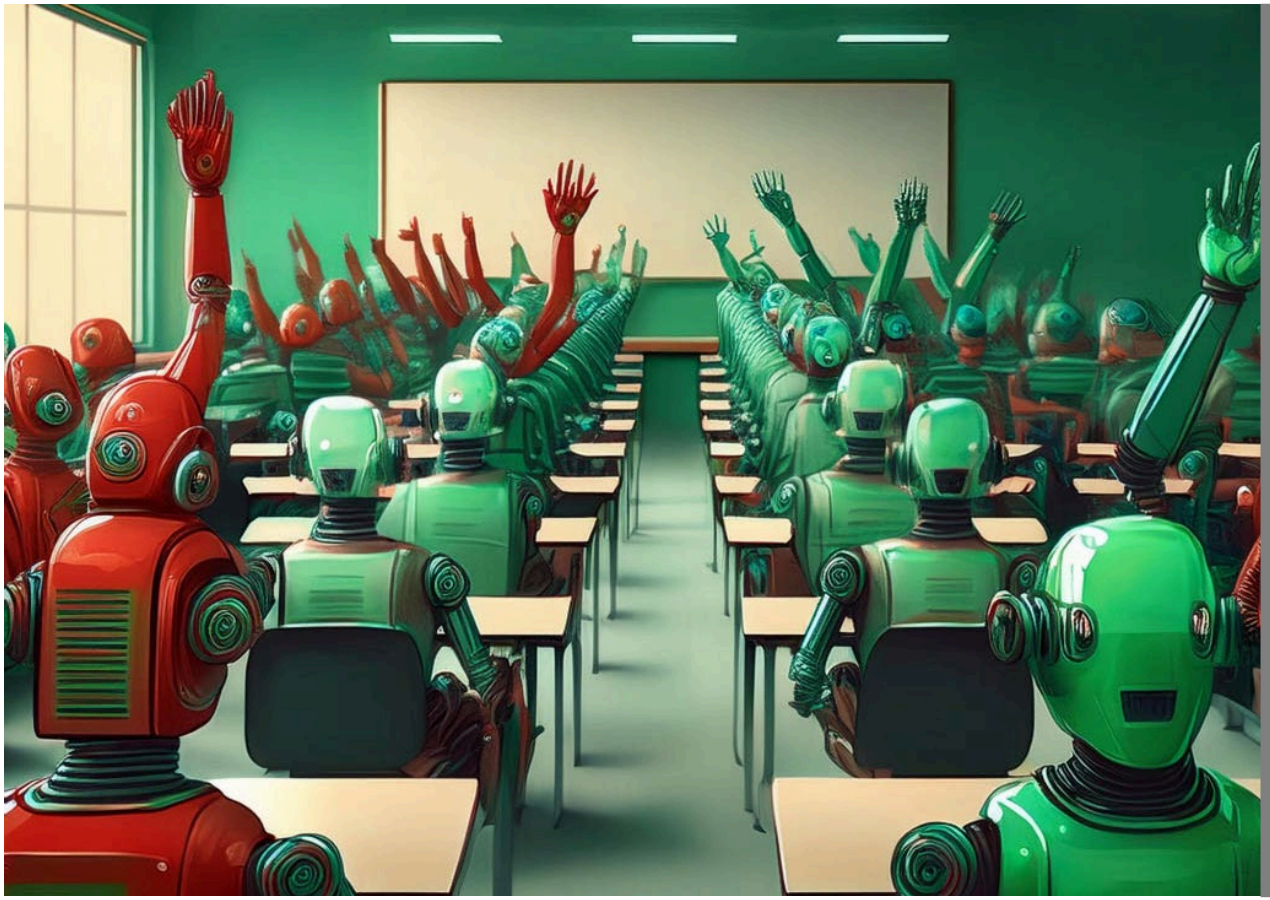
## The Intellyx Take

Conversions are the canaries in the coal mine of QoE. If there's anything wrong with the user experience, then conversions will take the hit.

Uncovering the cause of such issues, however, has long been a matter of guesswork. Is a slow server to blame? A network issue? Maybe a bug in the software?

The experience-centric operations that Conviva supports approaches such problems from the other side. Instead of looking at the behavior of individual technology components to uncover user issues, Conviva enables operators to begin with the user experience.

Only by focusing on each individual's experience as they travel on their journeys over time will any organization be able to understand and resolve issues that kill engagement and lead users to abandon processes mid-flight.



**By Jason English**

Director & Principal Analyst
Intellyx

# Operationalizing Full-Census Data to Optimize QoE

**Customers expect to get what they want, when they want it, at the lowest price.** They expect the freedom to choose between thousands of products and configurations, while at the same time receiving suggestions and content intuitively tailored to their needs. And if a problem arises, they have the agency to quickly take their business elsewhere.

Market leading digital businesses like Amazon and Netflix made dramatic market share gains by moving faster and orienting each new software feature and deployment around achieving customer outcomes at scale.

Unfortunately, most enterprises lack the insight to even get a handle on their existing technology stack to meet customer needs. They can try to correlate every metric and log emanating from their application estate in real time to make incremental performance improvements, but they'll still be missing the most important part: how to find causation between so much rich telemetry data and the key indicators of customer experience.

This new customer-centric reality sets the stage for *digital transformation*, where enterprises must align every aspect of their business operations around technology and data, to focus on efficiently meeting customer needs.

## Observability is disconnected from customer experience

As my colleague Jason Bloomberg points out in the previous article in this series, most observability platforms are not customer-centric. Instead of focusing on things like conversions and repeat business, they take a system-centric approach to data—gathering the 'golden signals' of latency, traffic, errors, and saturation—by looking at server logs, network packets, cloud usage metrics and the like.

This 'inward-out' approach to monitoring and analytics is useful for benchmarking the back-end impact of increased usage, or the network impact of regional cloud outages, or unanticipated changes to applications, or lost connections to downstream data services.

DevOps teams could even run programs that replay or synthetically simulate traffic, whether captured as browser request logs or RUM (real user monitoring), in order to test the scalability of clusters, database robustness, and API response times. All of these data are staples of optimizing IT operations, but they are tied to system events rather than specific customer experience.

Digital transformation requires an 'outward-in' approach that regroups observability data toward optimizing customer experience at every significant touchpoint, all the way down to small groups and individual users. Client-side data is captured at the diffuse endpoints customers are using: whether on a smartphone, digital TV, kiosk, home sensory network, or a field technician's tablet.

*"[Customer] experience occurs in fine-grained user cohorts. Checkout may be slow only for Android operating system users running version 2.4 of a retail store's app. There might be hundreds or thousands of users in an impacted cohort within a company's user base.*

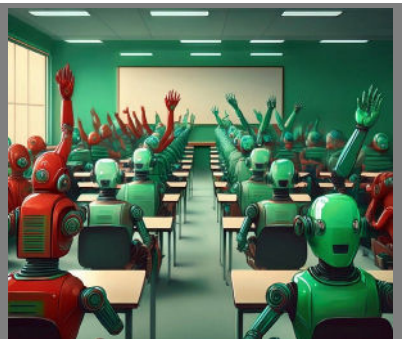
*"Finding these cohorts is a computational and AI problem that requires analysis across thousands of dimensions, such as all combinations of device models, app versions, geographical locations and other factors." – Aditya Ganjam, CPO & co-Founder, Conviva, in Forbes*

## The need for full-census comprehensive telemetry data

If system-centric observability demands full-stack telemetry data, then customer-centric observability demands ***full-census, comprehensive*** telemetry data.

Full-census telemetry ingests real-time event data from every user endpoint, from web browsers to Android and iPhone devices of every supported OS and hardware variety. Then, to be comprehensive, every action, every click, every switch between apps, developer tags in front-end code, and all of the relevant performance data of responses to each of those actions should be collected.

***In many cases, ingesting such comprehensive end-to-end telemetry will not only drive up cloud data costs, the load could even bog down the responsiveness of a customer's UI if client-side agents are churning away.***



To make things even harder, merely sampling incoming data to save costs or capacity could not possibly provide full-census, comprehensive telemetry. The critical moments that impact customer experience happen in the blink of an eye, irrespective of the intervals between samples. Still, that's too much data for any human to sort through.

## Transforming telemetry into actionable insights

Once full-census client data is in hand, it's time to start cleansing it. Similar to the data analytics world, this means metatagging each event for attributes and relevance beyond its unique ID, source, and time stamp, then filtering it so useful insights can be gleaned.

Then this data can be further correlated with system level data, whether derived from incoming client-side data or from any number of leading observability and SIEM platforms, or their related historical data lakes.

The Conviva Operational Data Platform contains several capabilities for ingesting, mapping data with metadata, and then filtering and aligning metrics to quality of experience for the enterprise.

For instance, a developer using a mobile toolkit from Adobe may insert that platform's instrumentation code on every client that it is deployed, which may generate similar 'pings' or flapping alerts everywhere—ok for checking status, but not always relevant to a developer's debugging process.

If a set of app users experiences a crash, the developer would want to match a missing status alert to specific devices with the crash event, rather than dig into all of the client instances that Adobe alerting code was deployed to.

Once we've reduced the signal to noise ratio, there's one critical question we need to ask of our data: "What was the customer trying to do at the time when an issue occurred?"

## Mapping state data to critical user flows

Critical user flows match the highest-value customer journeys enacted through an application. These dimensions are largely unique to each enterprise and its vertical industry. For a pharmacy app, it could be a prescription refill, or for a media provider, it could be the viewing of a purchased movie.

These critical user flows may happen within a React app UI a developer built for a device, but behind the scenes of each session, each step must maintain statefulness across several domains, from on-device functions such as camera inputs and storage, to third-party APIs, transaction providers, and data sources.

Conviva offers an AI-assisted approach to this problem, by offering real-time computation of stateful experience metrics from multiple data sources within a user flow, which are then semantically mapped to customer outcomes and preferences, which can be drawn from an extensive library of horizontal and vertical user flow templates or customized for the needs of specific cohorts of customers.

With this in hand, you can do new things, like breaking down a performance budget for specific high value user actions, such as a shopping cart checkout.

Let's say that you have a target of 2.5 seconds or less for a checkout payment interface to load, because current QoE indicators show that certain customers are abandoning carts if they have to wait 3 seconds or longer, and a specific cohort of Android 14.0 users are waiting much longer and occasionally timing out.

Looking at the performance data, developers are able to stratify the number of milliseconds each process within that checkout user flow takes. The API call to retrieve shipping info takes 1 second, but that third party service isn't under the developer's control. A price calculator takes 250ms, which is also under budget.

But for this Android 14.0 cohort, a cookie write has been disabled—causing transactions to hang for more than 2 seconds as they lose local state data. Time for developers to modernize to a modern session token method or secrets management!

Going forward, development and operations teams can operationalize full-census user data to bring new insights to the table when making critical decisions about which applications need improvement, and what new features to introduce based on their impact on the quality of experience of individual cohorts of users.

## The Intellyx Take

*"The customer is always right. If it isn't right, we'll make it right."*

Such maxims were often quoted by business leaders over the last 100-or-so years to signify how committed they were to customer service. Historically, if employees took these mission statements to heart and provided better service and quality products, their organizations would naturally benefit from more loyal customers.

But stating platitudes about customer focus or simply being data-driven won't create digital transformation.

Organizations that wait until customers are complaining to start gathering and filtering full-census and comprehensive client-side telemetry data into actionable and stateful quality of experience indicators are already losing customers to churn, and failing to convert some portion of prospective buyers due to a lack of insight.

Fortunately, if you are still in business, it's not too late to change the way you think about data in a customer-centric light. Even if you are already doing all the right things with data from a conventional observability and system availability point of view, just imagine the upside of being able to pinpoint what matters most to customers.

**By Jason Bloomberg**

Managing Director & Analyst
Intellyx

# Time-State Technology: The Key to Experience-Centric Operations

***Time-series data*** is a familiar concept in the observability world. Every time an application or a system adds a record to a database or log file, it includes a timestamp that represents the precise time it created that record.

The resulting table or file typically contains vast numbers of such records, each one representing some information at a particular point in time.

Querying this information is also straightforward and can answer questions, like 'what was the particular value of a field at some point in time', or 'what was the average value of a field over a specified time interval?'

Some important tasks, however, aren't nearly as straightforward. For example, asking a simple question like 'how long did it take a particular user to log into a web site?' would require a surprisingly complicated SQL query – one that not only takes an inordinate amount of time to craft, but is also likely to have a difficult-to-find bug.

In the first two articles in this series, my colleague Jason English and I defined the notion of 'quality of experience' (QoE) and discussed how to operationalize telemetry data to achieve this QoE.

However, there's a catch: point-in-time data is insufficient for evaluating QoE. Instead, organizations must leverage *continuous time*.

## Continuous time and time-state analytics

The shift from the traditional point-in-time, relational approach to time-series data to the continuous-time approach requires a rethink of how we organize and query data.

***Continuous time is the key to this rethink. Rather than breaking up time into discrete records with timestamps, we must think of time as a continuous entity.***

Instead of traditional database queries that provide filter and aggregation operations over tabular, time-stamped data, we need stateful, context-sensitive analyses over event streams with continuous time semantics.

The core abstraction of this new approach to data is the *timeline*. Instead of tracking point-in-time data, timelines must track changes of state of values in continuous time.

In other words, for our question about the time it takes to log into a web site, we have a set of events: the user begins the login process, the user hits the 'log in' button, the log in process is complete, and the user sees a web page showing that they are logged in.

Each of these events represents a change in state, and the user's QoE depends entirely on how much time it took to move through this sequence of events.

## Enumerating the timeline types

There are, in fact, three different ways for dynamic processes to vary over time. These timeline types include:

- *State dynamics* – each state has a value at each point in continuous time and changes at discrete points, as the example above illustrates.
- *Numerical values* – values that vary continuously over time, for example, how much time a user has to wait while a video feed buffers on their device.
- *Event data* – data that captures a sequence of discrete events, for example, user state updates, content delivery network (CDN) updates, and events that represent the user seeking additional data.

Based upon these timeline types, queries for time-state analytics intuitively follow timeline semantics, as the figure below illustrates:

```
1    SELECT TL_DurationWhere(
2      TL_LatestEventToState(playerStateChange) = 'buffer' AND
3      TL_HasExisted(playerStateChange = 'play') AND
4      NOT TL_HasExistedWithin(userAction = 'seek', 5s) AND
5      TL_LatestEventToState(cdnChange) = 'CDN1'
6    ) AS result
7    FROM heartbeats
8    TIMELINE WITH EVENT TIME t
9    EVALUATE AT EVENT TIME 2022-07-21 10:05:00
```

*Figure 1: A SQL-like query for time-state analytics (source: Conviva)*

The query in the figure above returns the total duration of buffering after play has started and includes only the buffering that is not immediately after the 'seek' action.

Without the timeline abstraction that leverages continuous time and time-state analytics, it would be impossible to build intuitive queries like this one that return the user's QoE at a point in time.

## Building the visual timeline abstraction

Relational data depends upon the geometric abstractions of logical data models and their tables. While these abstractions are well-understood (as is the structured query language that leverages them), they are inappropriate for representing timelines and the time-state analytics that extract useful information from them.

Instead, we need a new type of geometric abstraction that lends itself to timelines: the directed acyclic graph (DAG).

DAGs are a well-understood element of modern graph theory. A common example of a DAG is how formulas work in Excel spreadsheets: each formula performs a calculation that includes values in other cells, including cells that themselves have formulas in them. The set of all cells with formulas connecting them is a graph.

Such graphs can be arbitrarily complex, but they must follow one basic rule: they can't loop back on themselves. Make this mistake in Excel and you'll get an error. In other words, such graphs are both directed and acyclic.

We can express any end-to-end query on a timeline as a DAG. As a result, we can build a geometric compositional language based on DAGs that connect the three timeline types listed above.

Conviva, in fact, has built such a visual language it calls 'Stateful Event Analytics Made Easy,' or SEAM-EZ. The figure below illustrates a timeline-based metric written in SEAM-EZ:

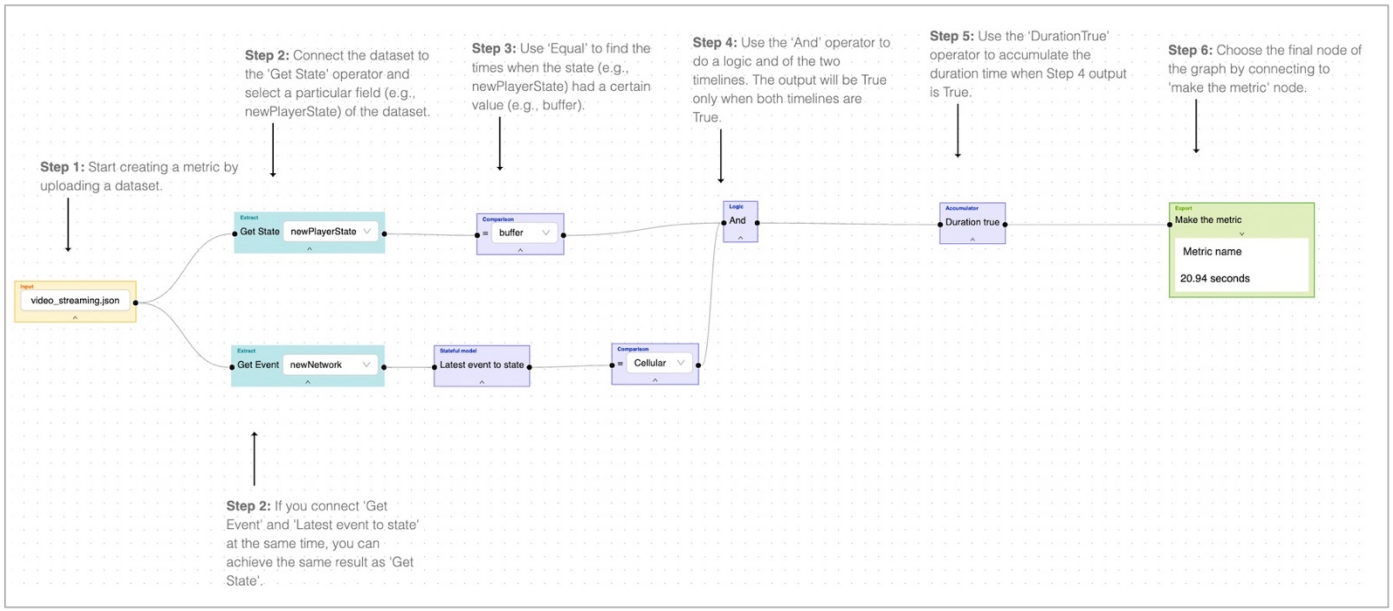



*Figure 2: A visual representation of a timeline DAG in SEAM-EZ (source: Conviva)*

The figure above illustrates how someone would define the buffering duration metric for a particular streaming channel, in this case cellular.

This intuitive no-code interface shows how simple it is to create DAGs that provide the stateful metrics organizations require to ensure that their users' QoE meets its goals.

## The Intellyx Take

Most operational tools are point-in-time: what is the log entry, metric value, or other telemetry at a particular moment.

User experience doesn't work this way, as it typically has a time component: how long did it take to log in? How long until a page or app responded?

Solving this problem requires a different way of structuring, tracking, and organizing operational data. The problem runs deep: the tabular abstraction underpinning relational databases and SQL falls short.

To solve the problem, Conviva has created a new timeline abstraction that represents all event stream data as timelines with a set of timeline operators to compute stateful metrics.

The proof is in the visual representation above. The underlying details are complex, but the geometric timeline abstraction offers a simple, but powerful approach for calculating the stateful metrics essential for QoE.

**By Jason English**

Director & Principal Analyst
Intellyx

# To Optimize the Customer Journey, Stay In the Flow

In this series, my colleague Jason Bloomberg and I have defined quality of experience' (QoE), discussed operationalizing telemetry data for QoE, and explored the dynamics of Time-State technology. All of which should help us break out of traditional thinking when optimizing how our critical customer-facing applications perform.

Unfortunately, all of the time-state telemetry data and quality-of-experience optima in the world can still produce suboptimal results, unless we can understand our customer's intentions and impressions during the moments they are taking action. Subjective indicators about the customer's state of mind make it very hard to extrapolate useful customer experience metrics.

Given this difficulty, it would be tempting to continue measuring the success of our customer-facing application in terms of concrete results, such as reduced customer churn, increased signups and sales revenue, or the old gold standard of high NPS (net promoter score) from happy customers who answer a quick pop-up survey to say they'd recommend us to a friend.

While useful for measuring company performance, results-oriented measures can be misleading for teams delivering a digital experience. For instance, if economic conditions improve for our customers, or we are lucky enough to carry a hot product everyone wants, more revenue would still give us little insight into what we should specifically improve about our application. We'd still be leaving money on the table.

That's why successful digital transformation doesn't focus on the destination—it focuses on improving the customer journey. And the lived experience of that journey consists of multiple 'customer flows' which are the cellular unit of measurement for experience-centric operations.

## Why are customer flows critical?

Flows are a familiar concept in software development. For instance, they have existed for years in web traffic monitoring, tracking the aggregated visitor click-throughs as they navigate a path through a website.

Observability dashboards may also display back-end flows, for instance, tracking sequences of API calls and responses, database reads and writes, REST transactions, or logs passing between services (i.e. the giant hairball map of system operations.)

Engineers are also looking at RUM (real user monitoring) or synthetic monitoring solutions–which can capture and/or help them replay the traffic patterns that influence application performance. Telemetry such as error rate, network traffic load, API response times, database latency, and so on can be isolated and captured.

Without customer flows, detailed measures are putting the cart before the horse. To collaborate effectively, we want development, operations, product management, sales, and support all monitoring the same flows as common actionable building blocks of customer experience.

## What's in a customer flow anyway?

As a basic unit of customer experience, you might think of each customer flow (or CF) as a subway or train station stop on the customer's journey where they spend moments or minutes to view information and/or make a decision.

Like previous stations visited on the customer journey, each CF has one or more entry points of other downstream activities that lead into it, and one or more exit points of upstream CF activities they could visit next, until they reach a terminus of success or failure.

Note, you can come up with your own concept of a CF, but I wouldn't recommend metaphors like 'lego blocks,' 'pipelines' or 'links in a chain' which would be useful for describing data movement and/or integration steps, rather than a unit of customer time and attention on their intended journey to a destination.

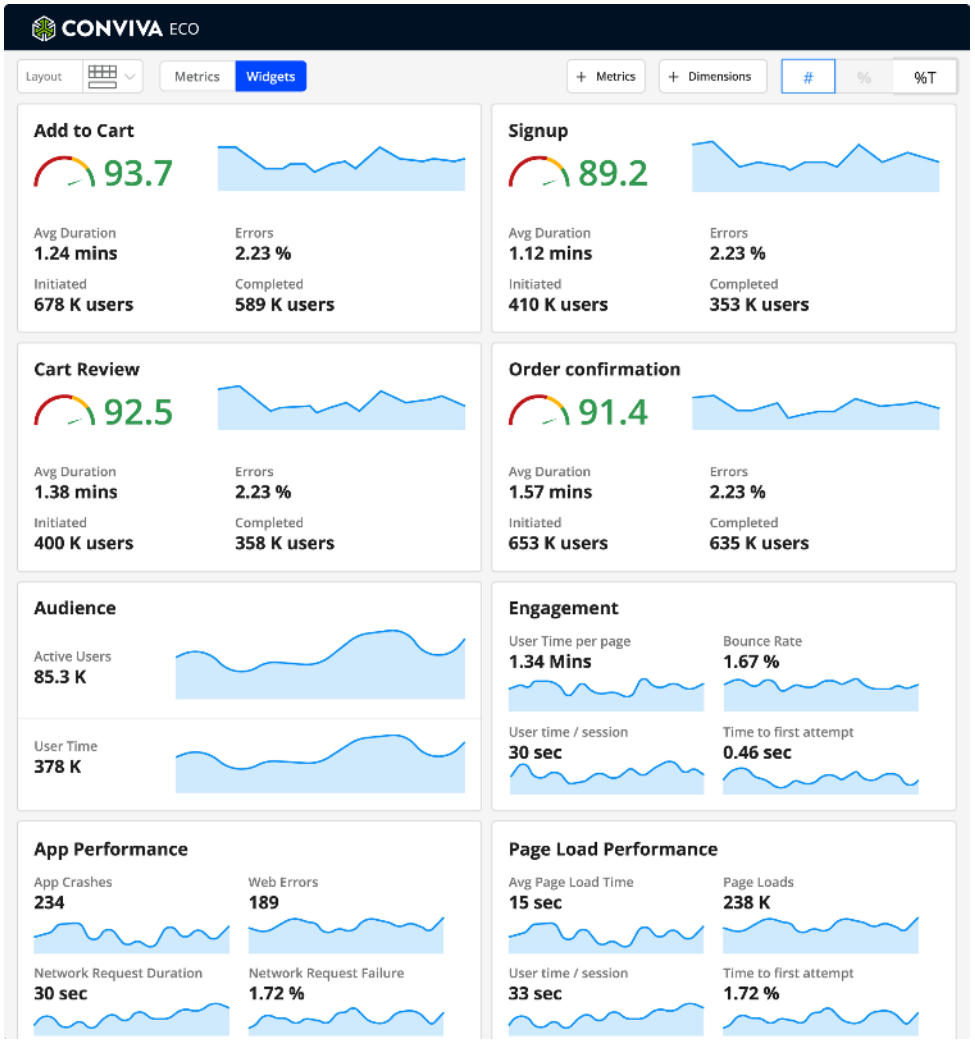



*Figure 1 (Right). Within this Conviva interface, customer flow metrics are presented as "widgets" or cards on a dashboard, each of which breaks down goals for time spent, errors, usage and success rates. Aggregated application performance metrics are shown under the flows.*

Are customers able to complete each flow, within an acceptable period of time, and without noticing any problems? Each flow needs to work perfectly to contribute to a great experience.

This requires that both product and engineering teams gain an end-to-end understanding of each customer journey, which can be broken down into critical customer "flows" such as "add user" "view shopping cart" and "check out."

By elevating flows as experience-centric metrics, we can look at them alongside conventional measurements. Both DevOps and business-side teams will collaborate more effectively by grouping granular system and traffic telemetry data underneath customer flows as a common unit of experience.

Teams can then set a baseline for what is acceptable, and set objectives for improvement toward an ideal, as they improve flows across all interactions between a customer and the digital brand over time.

## Solutions for the here and now

To make the example practical, let's try breaking down a typical e-retail scenario and getting into a customer flow.

What if customers are just abandoning carts in the middle of a checkout flow all of the sudden, even if there are no errors reported? Since the root cause of each particular abandonment at the Cart Review flow may not actually be reproducible, how can we figure out what caused it?

Looking at the aggregate observability data, we can see that the cart review data shows up on the page in under 1.2 seconds on average, with only a few anomalies, apparently due to bad wifi at the customer endpoint, so it doesn't seem to be a page load time issue.

This is where the value of full-census telemetry data from a solution like Conviva really comes in handy, as we drill into particular user sessions that abandoned their carts but did not have a network timeout.

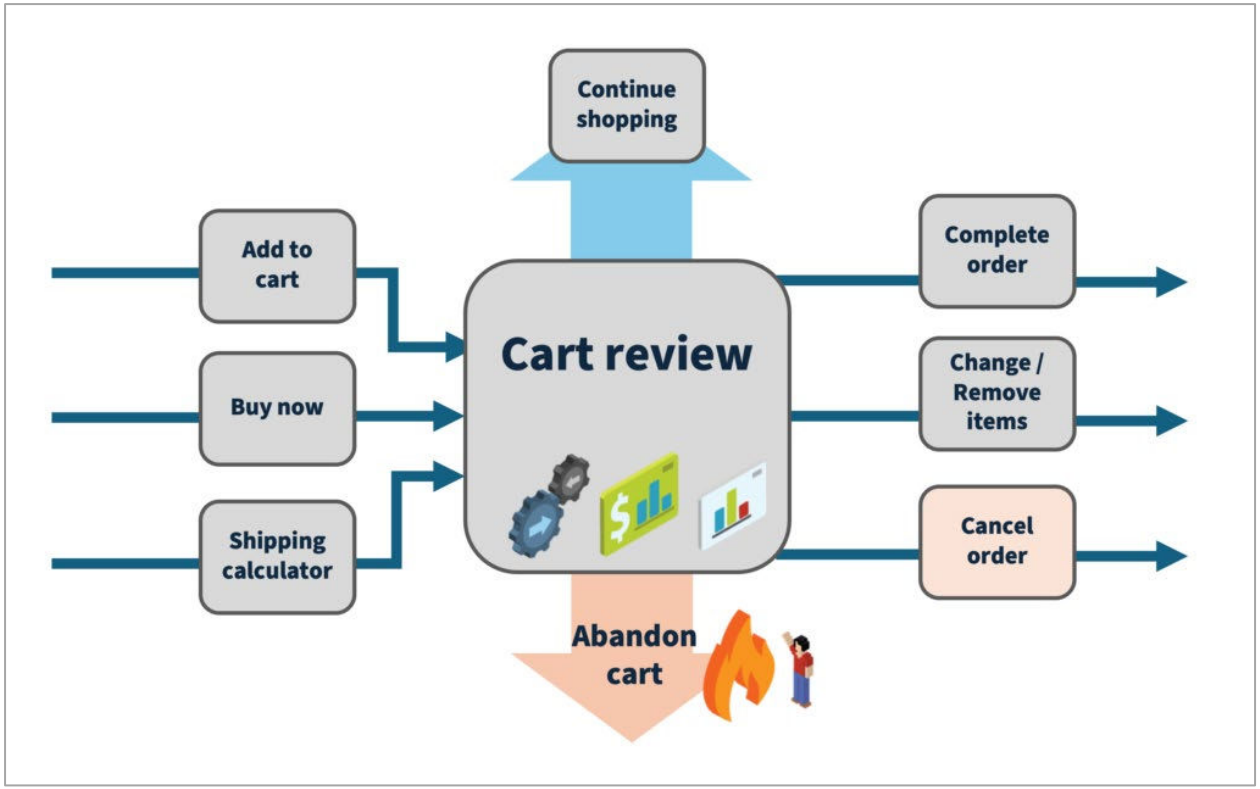



*Figure 2. A single customer flow for Cart Review with upstream and downstream customer flows. (Infographic by Jason English)*

Reviewing the smaller set of cart abandonments, we see that some are being presented with shipping costs, but some of the delivery dates are different, or fail to load for certain products in the cart.

Perhaps showing the estimated delivery date and shipping cost up front for the items in the cart might help resolve some of these issues.

To do that, developers also need to improve the performance of responses from logistics providers through the shipping API. A product manager could report a trouble ticket to those partners, telling them to pick up their own response times. That may work in some cases, but it doesn't take responsibility for the problem.

Instead, the engineering team can take some of that effort into their own hands to avoid dependencies, for instance by calculating and pre-loading the most likely or ideal shipping cost and time estimates next to the items in the cart. The estimated or cached delivery costs and times could then be delineated as 'guaranteed' at checkout time when the actual responses from vendors and shipping partners come in.

***If you have optimized all the flows as much as you can, and customers are still abandoning their carts, perhaps it is not an engineering problem at all, it is a design problem. Time to reduce required steps and delete some extraneous customer flows!***

## The Intellyx Take

Breaking down aggregated observability data into customer flows is not rocket science. In practice, it's more like regular math expressions—putting brackets around each customer action to determine the order of operations in a long equation.

You may have an array of telemetry data sources, observability tools, and service and incident management platforms at play in your application environment. What matters is what you do with that information to improve quality of experience. That represents your primary brand advantage.

Don't settle for a high-level overview of observability data, when you can live in the here and now with your customers and get in the customer flow.

## About the Analysts

**Jason Bloomberg** is Managing Director and Analyst of enterprise IT industry analysis firm Intellyx. He is a leading IT industry analyst, author, keynote speaker, and globally recognized expert on multiple disruptive trends in enterprise technology and digital transformation.

Mr. Bloomberg is the author or coauthor of five books, including *Low-Code for Dummies*, published in October 2019.

**Jason "JE" English** is Director & Principal Analyst at Intellyx. Drawing on expertise in designing, marketing and selling enterprise software and services, he is focused on covering how agile collaboration between customers, partners and employees accelerates innovation.

A writer and community builder with more than 25 years of experience in software dev/test, cloud and supply chain companies, JE led marketing efforts for the development, testing and virtualization software company ITKO from its bootstrap startup days, through a successful acquisition by CA in 2011. Follow him on Twitter at @bluefug.

## About Intellyx

Intellyx is the first and only industry analysis, advisory, and training firm focused on customer-driven, technology-empowered digital transformation for the enterprise. Covering every angle of enterprise IT from mainframes to cloud, process automation to artificial intelligence, our broad focus across technologies allows business executives and IT professionals to connect the dots on disruptive trends. Read and learn more at https://intellyx.com or follow them on LinkedIn.

## About Conviva

**Conviva** revolutionizes how businesses understand customer experience and engagement. Trusted by companies like Disney, NBC, and the NFL, Conviva is the first and best place to go to understand and optimize your customers' digital experiences to maximize satisfaction, conversion, and revenue. Our Operational Data Platform unlocks the power of full-census, comprehensive client-side telemetry—capturing all customer experiences across all devices and linking them to the performance of underlying services, in real time and at a fraction of the cost of alternative solutions.

.