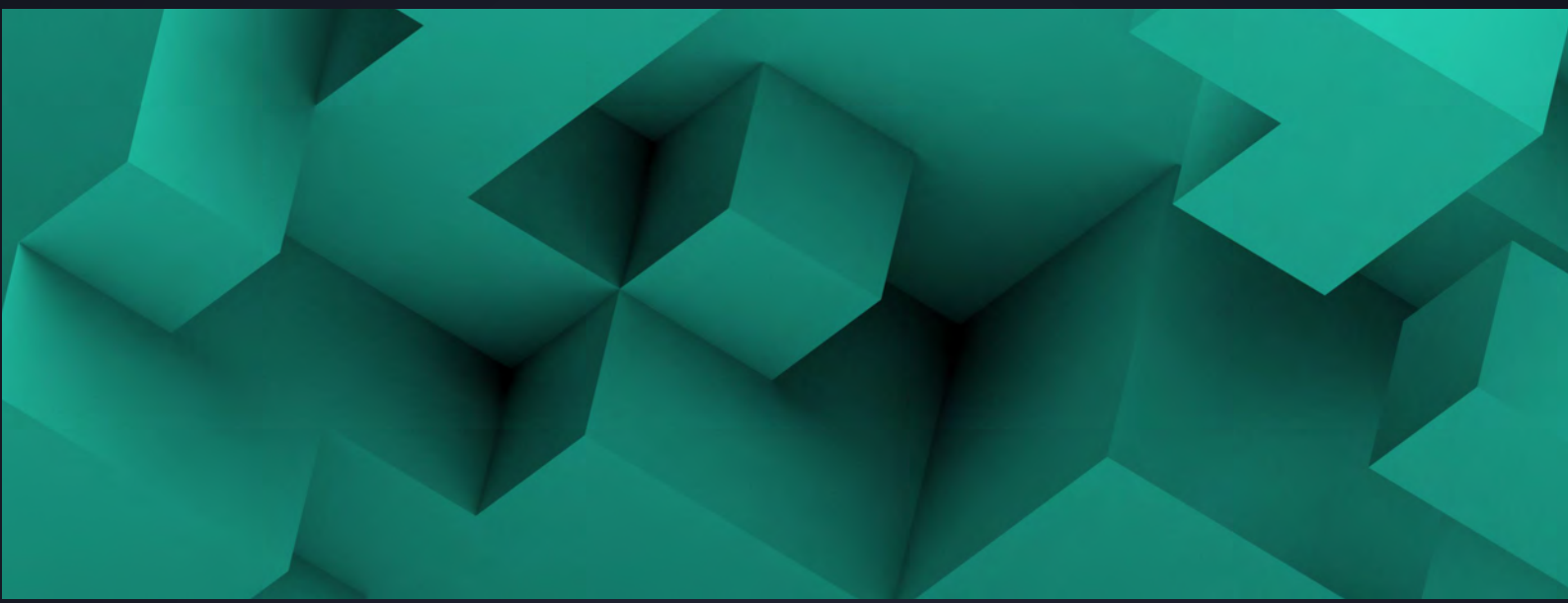
TALOS

ANALYSIS

Backtesting Risk Measures for Digital Assets



May 2026

TALOS

Backtesting Risk Measures for Digital Assets

Marco Marchioro
https://www.talos.com/
Version 0.91

May 2026*

Abstract

This paper presents a practical framework for backtesting hourly refreshed risk forecasts in digital-asset markets. We study one-day-ahead downside and upside tail measures produced by the Talos historical simulation engine, namely Value-at-Risk (VaR), Conditional Value-at-Risk (CVaR), Gain-at-Risk (GaR), and Conditional Gain-at-Risk (CGaR), across multiple confidence levels. The goal is operational: to verify that these risk figures remain reliable as new market data arrive.

The framework addresses calibration, stability through time, and comparative forecast accuracy. Calibration is assessed through joint identification functions for the threshold and tail mean, allowing frequency errors and tail-severity errors to be diagnosed separately. Comparative accuracy is evaluated with the Fissler–Ziegel FZ0 score and Diebold–Mariano tests. The same methodology is applied symmetrically to downside and upside measures. Because forecasts are updated hourly while the horizon remains one day, overlapping horizons induce serial dependence, making heteroskedasticity- and autocorrelation-consistent inference essential.

The paper also shows how the same risk measures can support one-hour outlier detection through normalized exceedance scores and simple rarity bounds. Overall, the methodology provides a unified and operationally transparent approach to monitoring digital-asset risk measures for tokens, derivatives, and portfolios.

1 Introduction and notation

Digital assets trade continuously on global venues. The ecosystem ranges from large-cap names such as Bitcoin and Ethereum to long-tail tokens with thinner liquidity and different microstructure. Institutional use of spot and derivative markets makes reliable tail-risk summaries essential for trading limits, treasury, and operational controls. Computing Value-at-Risk (VaR), Conditional Value-at-Risk (CVaR), and related figures is only the first step; those numbers must also be validated over time so that risk management stays aligned with realized outcomes.

Risk measures and gain measures are the key summary statistics produced by the Talos historical simulation engine. Their definitions, interpretation, and unified construction via finite-sample distortion weights are presented in a companion paper by Marco Marchioro, see reference [1]. In particular, that paper precisely defines the downside risk measures—VaR (*Value-at-Risk*) and CVaR (*Conditional Value-at-Risk*)—and their upside gain counterparts—GaR (*Gain-at-Risk*) and CGaR (*Conditional Gain-at-Risk*)—in a common framework. With an abuse of notation, in this paper we will often use the term *risk measures* to also denote the gain measures. Also, in the literature, most authors use the term *Expected Shortfall* (ES) for CVaR. We prefer to use the term *CVaR*,

*Latest review April 17, 2026

i.e. *conditional VaR*, instead of the more popular term *Expected Shortfall* because we are going to use it in conjunction with VaR in the context of backtesting.

This paper studies how to backtest risk measures for digital assets when forecasts are refreshed every hour. We consider downside measures, namely VaR and CVaR, and their upside counterparts, GaR and CGaR, at the confidence levels

$$\mathcal{L} = \{99.9\%,\ 99\%,\ 95\%,\ 50\%\},$$

i.e. in fractional notation $\ell \in \{0.999,\ 0.99,\ 0.95,\ 0.5\}$ in the formulas below. The Talos setting is operational rather than purely theoretical. For each token and each hour, a proprietary historical simulation engine produces *one-day-horizon* risk figures. The definitions and unified construction of those figures are in reference [1]. In actual computation Talos uses hourly market invariants and scaling to a daily-equivalent return to obtain the 1-day VaR/CVaR/GaR/CGaR numbers, following the technique of reference [2]. Here the question is different: once the numbers are produced, how do we check that they remain reliable when new market data arrive?

Backtesting is the answer to that question. In simple terms, backtesting compares forecasts made in the past with what happened next. A risk figure is useful only if it behaves well out of sample. A VaR forecast should be exceeded at about the right frequency. A CVaR forecast should describe the average size of losses when VaR is exceeded. The same idea applies on the gain side: a GaR forecast should be exceeded by unusually large gains at about the right rate, and CGaR should describe the average size of those gains.

The recent backtesting literature gives a useful way to organize this problem. One part is *calibration*: are the reported risk numbers statistically in line with realized outcomes? Another part is *comparative accuracy*: when two procedures are available, which one gives better forecasts out of sample?

The joint elicitability results of Fissler and Ziegel, as seen in references [6, 8, 9], make this framework available for the pair (VaR, CVaR), and hence also for the mirrored pair (GaR, CGaR). In this paper we use a short and practical version of that framework. For calibration, we rely on joint identification functions for the threshold and the tail mean. For comparative accuracy, we use the FZ0 score together with Diebold–Mariano tests. The same structure is applied to downside and upside measures. We then show how the same quantities can be used for outlier detection.

For ease of exposition, the running example is a single-asset portfolio of Bitcoin (BTC). The same identification functions, scores, and tests apply unchanged when the variables are the downside loss and upside gain of a digital-asset derivative—for example, a perpetual or an option—or of a *portfolio* of such derivatives (and spot positions), provided the Talos engine reports VaR/CVaR and GaR/CGaR for that P&L. The monitoring dashboards below use the same single-BTC framing; the methodology extends analogously to portfolios at every level.

1.1 Horizon, notation, and forecasts

We fix a *one-day* risk horizon and work in hourly time. Let $h = 24$ denote the number of hours in one day. At each hour t, the engine outputs forecasts (v_t, s_t, g_t, j_t), see table 1 for the definitions, for the same one-day downside loss and upside gain functionals as in reference [1]. The work in reference [2] develops hourly log-returns as approximate market invariants for digital assets, shows how to rescale from the 1-hour to the 1-day horizon, and uses the resulting daily-equivalent innovations to generate token price scenarios and daily-horizon risk measurement without an end-of-day close. We adopt that pipeline here: the risk measures Talos backtests are *1-day* objects, even though they are recomputed every hour.

For backtesting, the realized outcomes used in the identification functions and scores are *observed* $h = 24$ hours after the forecast was made. Concretely, the backtest pairs the forecast at time t with realized quantities that are fully known at time $t + h$.

Measure	Symbol	What it summarizes	Main backtesting question
VaR	v_t	Downside threshold at level ℓ	Are losses crossing the threshold at about the expected rate?
CVaR	s_t	Average downside severity beyond VaR	When losses cross VaR, is the average overshoot in line with the forecast tail mean?
GaR	g_t	Upside threshold at level ℓ	Are large gains crossing the threshold at the expected rate?
CGaR	j_t	Average upside severity beyond GaR	When gains cross GaR, is the average overshoot in line with the forecast tail mean?

Table 1: Main questions answered by the backtests.

For each token and each hour t, the risk engine produces four forecasts at each level $\ell \in \mathcal{L}$:

- downside threshold $v_t = \mathrm{VaR}_{\ell,t}$,
- downside tail mean $s_t = \mathrm{CVaR}_{\ell,t}$,
- upside threshold $g_t = \mathrm{GaR}_{\ell,t}$,
- upside tail mean $j_t = \mathrm{CGaR}_{\ell,t}$.

To keep notation light, the token index and the level ℓ are suppressed inside formulas. Every test below is run separately for each token and for each value of ℓ. When helpful, read "token" as any stand-alone risk object (single position, derivative book, or portfolio) for which the same forecasts and realized outcomes are available.

Let L_{t+h} denote the realized downside loss relevant for the VaR/CVaR forecast made at time t, and let G_{t+h} denote the realized upside gain relevant for the GaR/CGaR forecast made at time t. Both are tied to the one-day horizon and are observed at $t + h$. This paper does not assume any particular method for constructing L_{t+h} and G_{t+h} beyond consistency with reference [1] and the horizon convention above.

Table 1 summarizes the role of the four measures.

2 Backtesting goals and statistical tests

Forecasts are indexed by the hour t at which they are produced. The realized outcomes that enter the tests are L_{t+h} and G_{t+h} with $h = 24$ hours, i.e. they are observed one day after the forecast, as described in section 1.1. For production use, backtesting should answer three questions.

1. Are the risk figures *calibrated*? Calibration means that the realized data behave, on average, as the forecasts say they should. For VaR and GaR, this is mainly a frequency question. For CVaR and CGaR, it is a severity question.

2. Is the calibration stable *through time*? A model can look acceptable on a long sample and still fail in the periods that matter most, for example during stress, high volatility, or exchange-specific dislocations. Rolling windows and asset-level monitoring help separate persistent drift from short-lived episodes.

3. If more than one forecasting procedure is available, which one is *more accurate out of sample*? This question matters whenever the production engine is compared with a challenger model, a revised calibration, or a different implementation. A pure pass/fail test is not enough for this purpose; one needs a proper scoring rule that ranks forecasts by realized performance, as seen in references [8, 9].

These three questions matter at both the system level and the individual asset level. In a universe with hundreds of tokens, a panel summary can show broad shifts in model quality. But panel averages alone are not enough. A small number of consistently miscalibrated assets can be hidden within an otherwise acceptable overall result. Because risk limits are often set for each token separately, the most practical unit for monitoring is the individual asset.

2.1 Joint identification for VaR and CVaR

At level ℓ, define the VaR exceedance indicator

$$H_{t+h} = \mathbf{1}\{L_{t+h} > v_t\}.$$

For example, over a 30-day window of hourly forecasts, the sequence H_{t+h} is an array of 30×24 entries, each equal to 0 or 1. Under correct calibration, the expected exceedance rate is $1 - \ell$. A convenient joint calibration check for the pair (v_t, s_t) is given by the two identification functions

$$\psi_{1,t+h} = H_{t+h} - (1 - \ell),$$
$$\psi_{2,t+h} = v_t - s_t + \frac{1}{1-\ell}(L_{t+h} - v_t)H_{t+h}.$$

If the VaR and CVaR forecasts are correct, then both conditional expectations are zero:

$$\mathbb{E}[\psi_{1,t+h} \mid \mathcal{F}_t] = 0, \qquad \mathbb{E}[\psi_{2,t+h} \mid \mathcal{F}_t] = 0,$$

where $\mathcal{F}_t$ is the information set available at time t.

The first component checks threshold calibration. The second component checks whether the tail mean is consistent with the realized overshoot beyond VaR. This split is useful in practice. If the first component fails, the threshold itself is wrong. If the first component is near zero but the second is not, the threshold frequency may be acceptable while the tail severity is still misspecified.

2.2 Unconditional calibration tests

For a fixed token and level ℓ, let

$$\vec{\psi} = \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix} = \frac{1}{T} \sum_{t=1}^{T} \begin{bmatrix} \psi_{1,t+h} \\ \psi_{2,t+h} \end{bmatrix}.$$

It is well known that the sequences $\psi_{1,t+h}$ and $\psi_{2,t+h}$ are typically clustered in time (technically this condition is called heteroskedasticity) and that they are also autocorrelated. Hence, the naive sample covariance estimator may not be appropriate, since it does not take into account these two features.

Let $\widehat{\Sigma}$ be a heteroskedasticity- and autocorrelation-consistent covariance estimator, for example the Newey–West estimator, as seen in reference [7]. The unconditional calibration statistic is

$$W = T\, \vec{\psi}^{\top} \widehat{\Sigma}^{-1} \vec{\psi}, \tag{1}$$

which is asymptotically χ^2_2 under the null of correct calibration. In the following, we will use the acronym HAC to denote the heteroskedasticity- and autocorrelation-consistent properties.

In reporting, the joint statistic in (1) should be complemented by the component-wise t-statistics for ψ_1 and ψ_2. The joint test gives a clean pass/fail view, while the two components explain *why* the model is failing. This is important for monitoring and remediation.

2.3 Comparative accuracy

Calibration tells us whether a model is acceptable. It does not, by itself, rank two acceptable models. For model comparison we use the Fissler–Ziegel FZ0 score. On the loss scale,

$$S_\ell^{\mathrm{FZ0}}(v_t, s_t; L_{t+h}) = \frac{\mathbf{1}\{L_{t+h} > v_t\}}{(1-\ell)s_t}(L_{t+h} - v_t) + \frac{v_t}{s_t} + \log s_t - 1. \tag{2}$$

Lower average score means better joint forecasts for VaR and CVaR, as seen in reference [6] and reference [9].

Suppose that two procedures, say A and B, are available for the same token and level. Define the score difference

$$d_t = S_t^{(A)} - S_t^{(B)}.$$

The null of equal or better performance of model A against model B can be tested with a Diebold–Mariano statistic based on $\bar{d} = T^{-1}\sum_t d_t$ and a HAC standard error, as seen in reference [5]. This gives a simple and transparent ranking rule for challenger exercises.

2.4 Backtesting upside measures

The upside side of the distribution is treated in the same way as the loss side. This is useful for at least two reasons. First, many digital assets have strong skewness and intermittent upward jumps, so a full description of the distribution should be checked on both sides. Second, upside tail behavior is informative for strategies that are sensitive to large gains, short-covering episodes, or rapid repricing.

Let us define the upside exceedance indicator as

$$J_{t+h} = \mathbf{1}\{G_{t+h} > g_t\}$$

The joint identification functions for the pair (g_t, j_t) are

$$\phi_{1,t+h} = J_{t+h} - (1-\ell),$$
$$\phi_{2,t+h} = g_t - j_t + \frac{1}{1-\ell}(G_{t+h} - g_t)J_{t+h}.$$

The unconditional calibration tests are exact mirrors of the downside case after replacing $(\psi_1, \psi_2, L, v, s)$ by $(\phi_1, \phi_2, G, g, j)$.

For comparative accuracy, we use the mirrored FZ0 score

$$S_\ell^{\mathrm{FZ0,+}}(g_t, j_t; G_{t+h}) = \frac{\mathbf{1}\{G_{t+h} > g_t\}}{(1-\ell)j_t}(G_{t+h} - g_t) + \frac{g_t}{j_t} + \log j_t - 1. \tag{3}$$

Again, a lower average score means better joint forecasts.

The practical benefit of this symmetry is that the same code structure can be reused for both sides of the distribution. The only change is the input series and the corresponding pair of risk forecasts. This makes it easier to maintain a large-scale monitoring system across many tokens and many hours.

3 Practical implementation of hypothesis tests

This section translates the previous formulas into the objects computed in production. For each token and level, forecasts made at hours t are first aligned with realized one-day outcomes observed at $t+h$, and T denotes the number of valid forecast-realization pairs after this alignment. The main implementation point is that the backtest is updated every hour while the forecast horizon remains one day. Consecutive backtest pairs therefore use one-day windows that overlap for $h-1 = 23$ hours. Even if the underlying hourly innovations were independent, this overlap would already induce serial dependence in the calibration vectors and in the score differences. HAC inference is therefore the baseline throughout this paper.

3.1 HAC covariance with overlapping one-day horizons

After alignment, let $x_t \in \mathbb{R}^m$ denote the series whose mean is being tested. In the applications of this paper,

$$x_t = \begin{cases} (\psi_{1,t+h}, \psi_{2,t+h})^\top, & \text{downside joint calibration,} \\ (\phi_{1,t+h}, \phi_{2,t+h})^\top, & \text{upside joint calibration,} \\ d_t, & \text{Diebold–Mariano comparison.} \end{cases}$$

Write

$$\bar{x} = \frac{1}{T} \sum_{t=1}^{T} x_t.$$

For each lag $\ell = 0, 1, \ldots, q$, define the sample autocovariance

$$\widehat{\Gamma}_\ell = \frac{1}{T} \sum_{t=\ell+1}^{T} (x_t - \bar{x})(x_{t-\ell} - \bar{x})^\top.$$

With Bartlett weights

$$w_\ell = 1 - \frac{\ell}{q+1}, \qquad \ell = 1, \ldots, q,$$

the Newey–West estimator is

$$\widehat{V}_{\mathrm{NW}}(x) = \widehat{\Gamma}_0 + \sum_{\ell=1}^{q} w_\ell \big(\widehat{\Gamma}_\ell + \widehat{\Gamma}_\ell^\top\big). \tag{4}$$

This estimates the long-run covariance of $\bar{x} \cdot \sqrt{T}$. Hence the HAC covariance of the sample mean is

$$\widehat{\mathrm{Var}}_{\mathrm{HAC}}(\bar{x}) = \frac{1}{T} \widehat{V}_{\mathrm{NW}}(x).$$

In the scalar case, (4) reduces to

$$\widehat{\omega}_x = \widehat{\gamma}_0 + 2 \sum_{\ell=1}^{q} w_\ell \widehat{\gamma}_\ell,$$

where $\widehat{\gamma}_\ell$ is the sample autocovariance of the scalar series.

The bandwidth choice should reflect the hourly update frequency and the one-day horizon. Because successive one-day realizations overlap, a practical default is

$$q \geq h - 1 = 23.$$

We recommend $q = 48$, which corresponds to two days. If the empirical autocorrelation of x_t remains material beyond lag 48, a larger value should be used. Automatic bandwidth rules are possible, but the overlap structure already supplies a transparent lower bound and is easy to justify operationally [7]. Using Bartlett weights also guarantees a positive semi-definite estimator, as seen in reference [7].

3.2 Implementation of the joint Wald tests

For downside calibration, set

$$x_t = (\psi_{1,t+h}, \psi_{2,t+h})^\top, \qquad \bar{\psi} = \frac{1}{T} \sum_{t=1}^{T} x_t, \qquad \widehat{\Sigma}_\psi = \widehat{V}_{\mathrm{NW}}(x).$$

The null hypothesis is

$$H_0 : \mathbb{E}[\psi_{1,t+h}] = 0, \qquad \mathbb{E}[\psi_{2,t+h}] = 0.$$

The Wald statistic is

$$W_\psi = T\,\bar{\psi}^\top \widehat{\Sigma}_\psi^{-1} \bar{\psi}.$$

Under the null, W_ψ is asymptotically distributed as χ^2_2, and the reported p-value is

$$p_\psi = 1 - F_{\chi^2_2}(W_\psi).$$

In practice, acceptance or rejection is often based on a p-threshold p_0, for example $p_0 = 0.05$. Regardless of the threshold p_0, a rejection will be reported if the reported p-value is less than p_0.

The component-wise HAC t-statistics are

$$t_{\psi_i} = \frac{\bar{\psi}_i}{\sqrt{\widehat{\Sigma}_{\psi,ii}/T}} = \frac{\sqrt{T}\,\bar{\psi}_i}{\sqrt{\widehat{\Sigma}_{\psi,ii}}}, \qquad i = 1, 2.$$

These component statistics should always be reported together with the joint test. In operational use, t_{ψ_1} diagnoses threshold frequency, while t_{ψ_2} diagnoses tail severity.

The upside implementation is the exact mirror image. Set

$$x_t = (\phi_{1,t+h}, \phi_{2,t+h})^\top, \qquad \bar{\phi} = \frac{1}{T}\sum_{t=1}^{T} x_t, \qquad \widehat{\Sigma}_\phi = \widehat{V}_{\mathrm{NW}}(x),$$

and compute

$$W_\phi = T\,\bar{\phi}^\top \widehat{\Sigma}_\phi^{-1} \bar{\phi}, \qquad t_{\phi_i} = \frac{\sqrt{T}\,\bar{\phi}_i}{\sqrt{\widehat{\Sigma}_{\phi,ii}}}, \qquad i = 1, 2.$$

At the most extreme level, especially $\ell = 0.999$, short windows may contain very few exceedances for a single token. In that case the component means remain informative, but the joint Wald statistic can be unstable or simply too low-powered to be decisive. The practical remedy is to extend the window and to read the joint statistic together with the two component-wise diagnostics. Since we are using overlapping one-day horizons at an hourly frequency, an ideal backtesting window would span at least one year.

3.3 Implementation of the Diebold–Mariano comparison

For model comparison, let $S_t^{(A)}$ and $S_t^{(B)}$ denote the downside or upside FZ0 scores defined in equations (2) and (3), computed for the same token, level, and side of the distribution on the same aligned forecast-realization pairs. Define the score difference

$$d_t = S_t^{(A)} - S_t^{(B)}.$$

Because lower FZ0 scores are better, $\bar{d} > 0$ favors model B, while $\bar{d} < 0$ favors model A. Let

$$\bar{d} = \frac{1}{T}\sum_{t=1}^{T} d_t, \qquad \widehat{\omega}_d = \widehat{V}_{\mathrm{NW}}(d),$$

where $\widehat{V}_{\mathrm{NW}}(d)$ is the scalar Newey–West long-run variance computed with the same bandwidth rule $q \geq h - 1$. The Diebold–Mariano statistic is

$$t_{\mathrm{DM}} = \frac{\bar{d}}{\sqrt{\widehat{\omega}_d/T}}.$$

For a challenger exercise in which model B is compared against incumbent A, the natural one-sided null is

$$H_0 : \mathbb{E}[d_t] \leq 0 \qquad \text{against} \qquad H_1 : \mathbb{E}[d_t] > 0.$$

When $\mathbb{E}[d_t] = 0$, t_{DM} is asymptotically standard normal [5]. For a non-directional comparison of the two models, report a two-sided p-value instead of the one-sided rule above.

In production, the implementation is uniform across all tests: align forecasts with outcomes at $t + h$, build the relevant series x_t or d_t, estimate the long-run covariance using the Newey–West estimator with $q \geq 23$, with a suggested value of $q = 48$, and then compute the corresponding Wald or Diebold–Mariano statistic. The only ingredients that change across applications are the identification functions or the score differences; the inference layer is the same throughout.

4 Outlier detection at the one-hour horizon

Outlier detection is closely related to backtesting, with some important differences. Backtesting asks whether the risk forecasts are good *on average*. Outlier detection asks whether a *particular realized move* is unusually large relative to the forecast made just before it happened.

One-hour horizon for outliers. The preceding sections use a *one-day* horizon and a 24-hour observation lag for backtesting. For *outlier* screening, Talos uses a *one-hour* horizon instead: the realized loss and gain are those over the single hour $(t, t + 1]$, observed at $t + 1$. In this section, v_t, s_t, g_t, and j_t denote the corresponding *one-hour* VaR/CVaR/GaR/CGaR forecasts at time t, and L_{t+1} and G_{t+1} denote the realized downside loss and upside gain over that hour. The symbols are reused for brevity; the horizon (one day vs. one hour) should always be clear from context.

The risk measures provide a natural way to define time-varying, asset-specific outlier thresholds. For downside moves, a first alert is simply a VaR exceedance:

$$L_{t+1} > v_t.$$

This is a rare event when ℓ is high, but it is not always a severe one. On the loss side, severity is captured by the normalized overshoot,

$$z_{t+1}^{\mathrm{loss}} = \frac{(L_{t+1} - v_t)^+}{s_t - v_t}, \tag{5}$$

Similarly, for the gain side we have

$$z_{t+1}^{\mathrm{gain}} = \frac{(G_{t+1} - g_t)^+}{j_t - g_t}, \tag{6}$$

where $(x)^+ = \max(x, 0)$.

When $z_{t+1}^{\mathrm{loss}} \approx 1$, the realized loss is roughly the average tail event beyond VaR. Similarly, when $z_{t+1}^{\mathrm{gain}} \approx 1$, the realized gain is roughly the average tail event beyond GaR. Values much larger than 1, for either side, indicate unusually severe downside or upside outliers.

These z-scores are useful because they convert raw token moves into a scale that is already adjusted for the current risk regime. A 7% hourly move may be routine for one token and exceptional for another. The scores in equations (5) and (6) normalize the move by the forecast tail width rather than by a fixed absolute threshold.

A second advantage is that VaR/CVaR and GaR/CGaR provide simple rarity bounds for directional outliers. Under correct calibration,

$$\mathbb{P}(L_{t+1} \geq v_t + \kappa(s_t - v_t) \mid \mathcal{F}_t) \leq \frac{1 - \ell}{\kappa}, \qquad \kappa > 0,$$

and similarly,

$$\mathbb{P}(G_{t+1} \geq g_t + \kappa(j_t - g_t) \mid \mathcal{F}_t) \leq \frac{1-\ell}{\kappa}.$$

These bounds follow from Markov's inequality applied to the overshoot beyond the threshold. They are conservative, but operationally useful: choosing κ sets a clear severity tier.

For example, one may use the following logic for each token and each hour:

- **alert**: $L_{t+1} > v_t$ or $G_{t+1} > g_t$,
- **strong outlier**: $z^{\text{loss}}_{t+1} \geq 1$ or $z^{\text{gain}}_{t+1} \geq 1$,
- **severe outlier**: $z^{\text{loss}}_{t+1} \geq 3$ or $z^{\text{gain}}_{t+1} \geq 3$.

The exact thresholds can be adapted to the use case. The important point is that the thresholds are dynamic and grounded in the same forecast distribution that is being monitored.

In practice, an outlier flag should trigger a second-stage review rather than an automatic conclusion. A flagged event can represent a genuine market shock, a token-specific news event, a market microstructure issue, or a data problem such as a stale or corrupted print. Backtesting helps keep this process disciplined: if the risk figures are well calibrated, the false-alarm rate remains under control.

5 Operational guidance for hourly monitoring

5.1 Token-level dashboards and governance

In a large token universe, pooled results are informative but not sufficient. A few large and liquid assets can dominate the picture, while smaller assets fail silently. For each token and level ℓ, the dashboard reports the rolling mean of the two calibration components, the joint test statistics, and the recent exceedance history. Cross-sectional summaries are best treated as a second layer that complements the asset-level view.

This point is especially important for single assets because a user often consumes the risk figure in a local way: the question is not whether the engine is broadly acceptable, but whether the number shown today for a given token is trustworthy. A persistent drift in the backtest of one token is already enough to justify investigation, even if the panel average remains stable. The same idea applies to a dedicated book of digital-asset derivatives or to a portfolio of such positions when the risk measures are produced on the resulting P&L series.

5.2 Interpreting multiple confidence levels

The four confidence levels in $\mathcal{L}$ serve different purposes. The very high levels, especially 99.9% ($\ell = 0.999$), focus on rare tail events and are the most relevant for stress-oriented controls. The levels 99% and 95% provide more statistical power and tend to reveal problems earlier. The level 50% ($\ell = 0.5$) is different: it is not an extreme-tail diagnostic. We keep it because it acts as a reference point for the middle of the conditional distribution.

This reference is useful in interpretation. If the level 50% is well calibrated but the high levels are not, the issue is likely concentrated in the tails. If both central and extreme levels fail, the problem is broader and may reflect location, scale, or distributional shape. In that sense, the collection of levels gives a simple map of *where* the model is failing.

5.3 Estimation and engineering choices

A few implementation choices are important in hourly digital-asset data.

First, serial dependence should be expected, both because returns themselves may show dependence and because model updates are performed every hour while the risk horizon remains one day

and the observation lag remains $h = 24$ hours. HAC covariance estimators are therefore preferable to i.i.d. standard errors, as seen in reference [7].

Second, no-crossing constraints should be enforced in production: on the downside one should have $s_t \geq v_t$, and on the upside $j_t \geq g_t$. Otherwise the interpretation of the tail means becomes unstable, and both the identification functions and the FZ0 score may behave poorly. Therefore, the dashboard should raise a flag when $s_t \simeq v_t$ or $j_t \simeq g_t$. In these cases, the tail mean is not well-defined and we might not be able to draw conclusions about outlier detection.

Third, the most extreme level accumulates evidence slowly at the single-asset level. For this reason, rolling windows and severity measures should be read together. A single statistic is rarely sufficient for operational judgment.

6 Conclusion

The proposed methodology is deliberately narrow. It does not try to exhaust the full backtesting literature. Instead, it focuses on one coherent framework that fits the present use case well.

The main advantage of the joint identification approach is interpretability. It separates threshold calibration from tail-severity calibration in a way that is easy to explain to non-specialists. The first component says whether we cross the threshold too often or too rarely. The second says whether the average size of the tail events matches the reported tail mean. This is more informative than a pure exceedance count.

The main advantage of the FZ0 score is that it gives a clean answer when two procedures must be compared. This is relevant whenever the production historical simulation is revised, simplified, or benchmarked against an alternative engine. Traditional pass/fail backtests are not enough for this task because they do not provide a consistent ranking of forecast quality, as seen in reference [8].

Alternative backtests for expected shortfall and related tail measures do exist, as seen in references [3, 4]. We do not use them here because Talos's main goal is to monitor a paired threshold-and-tail-mean forecast, both on the downside and on the upside, with one uniform implementation. For that goal, the joint framework is a natural fit.

In a future paper we will present concrete numerical results on how the framework developed here can be applied to monitor the quality of the risk measures computed in Talos production systems.

References

[1] Marco Marchioro, *Finite-Sample Distortion Measures: Unified Risk and Gain via Scenario Weights*, Talos Quant Research, https://www.talos.com/insights, February 2026. 1, 2, 3

[2] Marco Marchioro, *Hourly Market Invariants for Price Simulations in Digital-Asset Markets*, Talos Quant Research, https://www.talos.com/insights, *Working paper*, Second Quarter 2026. 2

[3] Acerbi, C. and B. Szekely (2014). Back-testing expected shortfall. *Risk*, 27, 76–81. 10

[4] Bayer, S. and T. Dimitriadis (2022). Regression-based expected shortfall backtesting. *Journal of Financial Econometrics*, 20(3), 437–471. 10

[5] Diebold, F. X. and R. S. Mariano (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3), 253–263. 5, 8

[6] Fissler, T. and J. F. Ziegel (2016). Higher order elicitability and Osband's principle. *The Annals of Statistics*, 44(4), 1680–1707. 2, 5

[7] Newey, W. K. and K. D. West (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3), 703–708. 4, 6, 10

[8] Nolde, N. and J. F. Ziegel (2017). Elicitability and backtesting: Perspectives for banking regulation. *The Annals of Applied Statistics*, 11(4), 1833–1874. 2, 4, 10

[9] Patton, A. J., J. F. Ziegel and R. Chen (2019). Dynamic semiparametric models for expected shortfall (and Value-at-Risk). *Journal of Econometrics*, 211(2), 388–413. 2, 4, 5

[10] Rockafellar, R. T. and S. Uryasev (2000). Optimization of conditional value-at-risk. *Journal of Risk*, 2(3), 21–41.

talos.com