



010101010010101010  
110101010 110101010  
101010101 101010101  
010101001010101001  
010101010010101010  
110101010 110101010  
101010101 101010101  
010101001010101001  
010101010010101010  
110101010 110101010  
101010101 101010101  
010101001010101001

# Case Study: Turning a 4X Price Hike into a 20% Cost Reduction

How a leading transportation enterprise used independent tokenization to regain control over payment processing.



# Executive Summary

When a major transportation enterprise was hit with a sudden 400% increase in payment processing fees, it faced a critical financial decision point. Fortunately, the company had already taken proactive steps to control its payment data by implementing DataStealth's independent tokenization solution well before the threat

emerged. Originally deployed to reduce PCI DSS compliance scope, this architecture ultimately gave the enterprise full control over its tokenized data. The result: a seamless switch to a new processor at a 20% lower rate, zero customer disruption, and millions in annual savings.



## Customer Profile

This enterprise is a market leader in the transportation sector, managing critical infrastructure and serving millions of customers every year. With thousands of daily payment transactions, even small shifts in payment processing costs can translate into multi-million-dollar swings in annual expenditures.

The company's leadership recognized early that controlling payment data wasn't just about compliance, it was a strategic business necessity. They sought a future-proof architecture that would reduce risk, lower costs, and keep them flexible in a rapidly shifting payment ecosystem.



## The Challenge: A Sudden 400% Fee Increase

Without warning, the company's incumbent payment processor - recently acquired by a larger entity - announced a 400% increase in transaction processing fees.

For many enterprises, this type of ultimatum creates an impossible situation:

Accept inflated pricing,

# OR

Face expensive migration efforts that could involve:

- Paying **millions** in break fees to retrieve tokenized data,
- Asking customers to re-enter payment details (**risking 20% churn**),
- Navigating complex and costly IT integration projects.



But this enterprise was in a uniquely **strong** position.

# The Strategic Advantage: Independent Tokenization with DataStealth

Months before the processor's price hike, the transportation company had deployed DataStealth's vaulted tokenization platform – not for pricing leverage, but to minimize PCI DSS scope and reduce audit burdens.

**This decision turned out to be transformative.**

With DataStealth controlling the token vault, the company, not their payment processor, owned the relationship to their tokenized payment data. This gave them total portability and negotiating leverage, enabling them to bypass break fees, avoid customer churn, and switch providers with ease.

20%



## How It Works:

### Full Control Over Tokenized Payment Data

- Tokenization happens before any card data enters the company's environment.
- Detokenization occurs only when data is passed to a processor.
- Tokens are stored in the DataStealth vault, not with any third-party processor.

This architecture ensures:

- Payment data remains out of scope for PCI DSS,
- The merchant never holds or sees cardholder data (PAN),
- **The enterprise can route payment transactions to any processor, with no technical rebuild required.**

## Results:

### Strategic Flexibility, Financial Wins

#### 20% Reduction in Processing Costs

Instead of accepting a 4X fee hike, the enterprise issued an RFP and secured a 20% lower rate with a new processor.

The migration cost was negligible. The ROI on the switch was realized in weeks, with long-term savings projected in the tens of millions.

#### Zero Disruption to Customers

Because DataStealth held the tokens, the switch required no customer re-enrollment. The risk of payment abandonment and churn was completely avoided, protecting customer experience and revenue.

#### No Break Fees, No Migration Costs

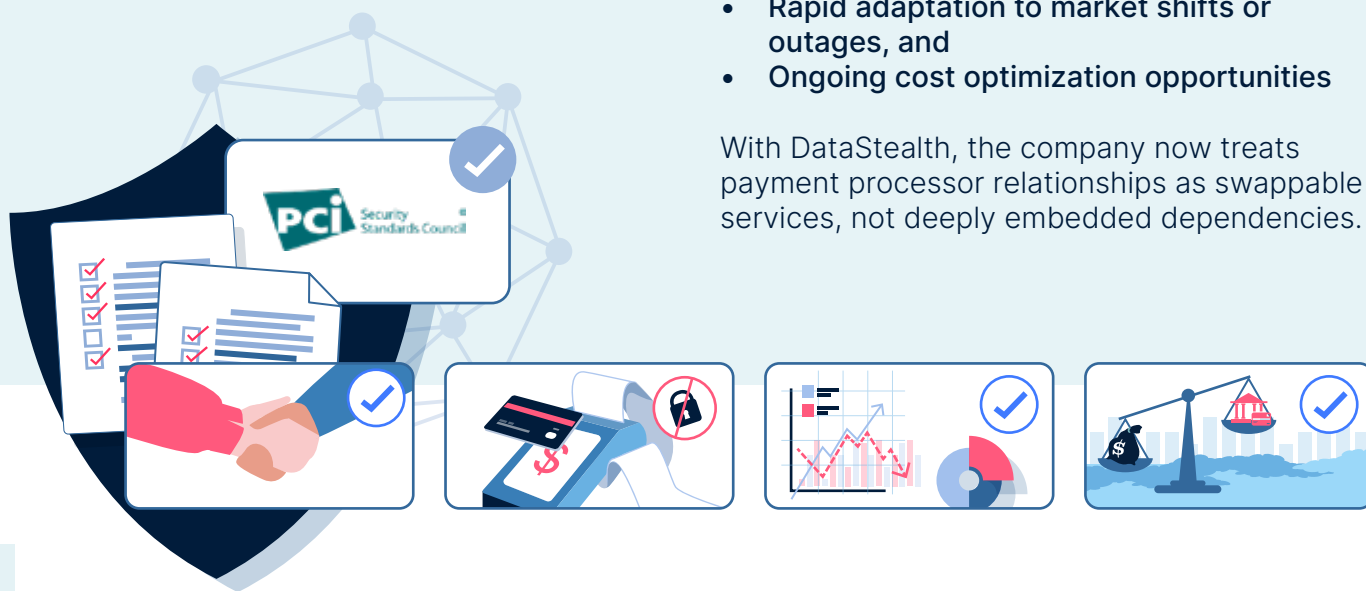
Since the token vault was independent, there were no break fees. Switching processors was a configuration update, not a complex migration project.

## Strategic Impact: Long-Term Agility and Control

By implementing DataStealth as part of its PCI DSS compliance strategy, the company gained much more than audit relief:

- Negotiation power with any processor,
- Freedom from vendor lock-in,
- Rapid adaptation to market shifts or outages, and
- Ongoing cost optimization opportunities

With DataStealth, the company now treats payment processor relationships as swappable services, not deeply embedded dependencies.



## Key Takeaways

Control Over Tokenized Data is Control Over Your Costs.

### Transform compliance into strategic advantage

What started as a PCI scope reduction project gave the enterprise the power to avoid a pricing trap and reduce costs.

### Achieve payment portability without disruption

Customers experienced no change and the company avoided costly break fees and a 20% churn risk.

### Operational agility through smart architecture

Switching processors is now a simple IT task, not a multimillion-dollar project.

### Future-proof your payment strategy

Payment processors can be changed, renegotiated, or replaced—on the enterprise's timeline, not the vendor's.



# Is Your Payment Data an Asset or a Liability?

This transportation enterprise didn't wait for a pricing crisis to take action. By prioritizing data control and PCI efficiency, they unlocked long-term cost savings and operational freedom.

With DataStealth, they turned a potential 400% fee increase into a 20% recurring discount while preserving full business continuity.

## Book a Demo Today

DataStealth empowers your business to:

- Eliminate payment vendor lock-in,
- Slash processing costs,
- Protect customer experience,
- Minimize PCI DSS scope, and
- Regain strategic leverage over payment operations

Take control of your payment data **before** someone else does.

Contact us today to explore your savings potential.

