GENERAL

mt 195 swift message example

Published: September 3, 2026 | Index ID: #-

Introduction

In the world of international finance, the SWIFT network is the backbone that facilitates secure, standardized communication between banks and financial institutions. Among the myriad of message types that SWIFT supports, the MT 195 stands out as a crucial tool for managing payment orders and related instructions. Whether you are a compliance officer, a treasury manager, or a software developer working on financial integrations, understanding how to read, construct, and troubleshoot an MT 195 message is essential. This article dives deep into the anatomy of the MT 195, walks through a detailed example, and provides practical guidance on best practices and common pitfalls.

What is MT 195?

Definition and Purpose

MT 195 is a specific type of SWIFT MT (Message Type) used primarily for “Payment Order – Return” and “Payment Order – Modification” messages. It enables a financial institution to communicate changes to previously sent payment orders or to return a payment that cannot be processed for various reasons, such as insufficient funds or regulatory constraints.

History and Evolution

The MT 195 was introduced in the early 1990s as part of SWIFT’s effort to streamline payment processing across borders. Over the years, its structure has remained largely stable, but updates to the SWIFT standards and the introduction of newer message types (e.g., MT 103 for single customer credit transfers) have refined its use cases.

Where MT 195 is Used

- When a bank needs to reverse or modify a customer’s payment order.
- To notify the sending institution of a failed transaction.
- In internal reconciliation processes where payment details need to be updated.

Understanding SWIFT Messages

SWIFT Network Overview

The Society for Worldwide Interbank Financial Telecommunication (SWIFT) provides a secure messaging platform that allows banks to exchange financial information quickly and accurately. Messages are encoded in a standardized format, ensuring that each party interprets the data consistently.

Message Types and Formats

SWIFT messages are categorized into three main families: MT (Message Type) for financial transactions, MX (XML) for newer messages, and MX (XML) for non-financial messages. MT messages are further divided into subtypes, each with a specific purpose. MT 195 belongs to the “Customer Credit Transfer” family.

Why MT 195 Matters

In a global payment ecosystem, errors or delays can have significant financial and reputational consequences. MT

195 provides a standardized mechanism for correcting or canceling payments, reducing the likelihood of settlement failures and improving overall operational efficiency.

Structure of MT 195

Basic Format

An MT 195 message is composed of five blocks:

1. Block 1 – Basic Header
2. Block 2 – Application Header
3. Block 3 – User Header
4. Block 4 – Text Block
5. Block 5 – Trailer

Each block contains specific fields that convey distinct pieces of information, from the originating institution to transaction details.

Block 1 – Basic Header

This block identifies the message's origin and destination, along with a timestamp. It follows the format:

{1:F01BANKBEBBAXXX0000000000}

Where **F01** indicates the message type and **BANKBEBBAXXX** is the BIC of the sending bank.

Block 2 – Application Header

The application header specifies the message type (e.g., 195) and the intended recipient. Example:

{2:O1951234567890BANKDEFFXXXXN}

Here, **O195** denotes an outgoing MT 195, and **1234567890** is the message reference.

Block 3 – User Header

Optional fields such as **5A** (Sender's Reference) or **5B**(Receiver's Reference) can appear here. They are often used for internal tracking.

Block 4 – Text Block

Block 4 contains the core transaction data, organized into numbered fields (e.g., **20** for the transaction reference, **32A** for the amount and currency). Each field may span multiple lines.

Block 5 – Trailer

The trailer holds message authentication information, such as a digital signature, ensuring integrity.

MT 195 Message Example

Sample Message

Below is a fully formatted MT 195 message that illustrates a typical return scenario:

{1:F01BANKBEBBAXXX0000000000}
{2:O1951234567890BANKDEFFXXXXN}
{3:{103:123456789}}
{4:

:20:123456789
:21:987654321
:32A:210801EUR1000,00
:33B:EUR1000,00
:50K:/1234567890
JOHN DOE
123 MAIN STREET
BRUSSELS
:59:/987654321
JANE SMITH
456 HIGH STREET
BERLIN
:71A:OUR
:72:/TRN/123456789
:73A:RETURN
:73B:INSUFFICIENT FUNDS
:73C:REASON
}
{5:{CHK:ABCD1234EFGH5678}}
Field Breakdown

- 20 – Transaction Reference Number
- 21 – Related Reference (e.g., original payment reference)
- 32A – Value Date, Currency, Amount
- 33B – Currency and Amount (used in certain contexts)
- 50K – Ordering Customer
- 59 – Beneficiary Customer
- 71A – Details of Charges
- 72 – Sender to Receiver Information
- 73A – Return Code
- 73B – Return Reason
- 73C – Additional Return Details

Common Variations

While the structure remains consistent, the specific fields used can vary depending on the institution's internal processes and regulatory requirements. For instance, some banks include field **72A** to provide additional transaction details, or field **73D** for detailed return codes.

How to Read and Interpret MT 195

Decoding Field Numbers

Each field number in Block 4 has a predefined meaning. Familiarity with the SWIFT MT 195 field list is essential. Resources such as SWIFT's official documentation or the **SWIFT Field Reference**guide can help you decode each field accurately.

Interpreting Data Elements

Data elements often contain composite information. For example, field **32A** packs the value date, currency, and

amount into a single string: **210801EUR1000,00**. Understanding the format (YYMMDD for the date, three-letter ISO currency code, and a comma as the decimal separator) allows you to parse the data programmatically.

Typical Use Cases

- Reversing a payment due to an error in the original amount.
- Returning a payment because the beneficiary's account was closed.
- Modifying the beneficiary's details mid transaction.

Creating MT 195 Messages

Manual Construction

For small volumes or testing, you can craft MT 195 messages manually using a text editor. Ensure that each block is correctly delimited with curly braces and that line breaks are preserved.

Using SWIFT Tools

Many banks employ SWIFT's **SWIFTNet MT** or third party software such as **Open Swift** to generate messages automatically. These tools validate the syntax against SWIFT standards, reducing the risk of errors.

Validation and Testing

Before sending a message to the live network, perform a series of tests:

1. Validate the message against the MT 195 Field List.
2. Check that mandatory fields are present.
3. Run a simulation in a sandbox environment.
4. Verify that the receiving institution acknowledges receipt.

Common Errors and Troubleshooting

Syntax Issues

Typical syntax errors include missing curly braces, incorrect field separators, or invalid field numbers. These errors usually result in a **SWIFT error code** such as **0.0.1.0** (Invalid block 4). Use a SWIFT validator to pinpoint the exact location.

Data Consistency Problems

Inconsistent data, such as a mismatched currency code between fields **32A** and **33B**, can trigger rejection. Always cross check amounts and currencies across related fields.

How to Resolve

- Re validate the message using a SWIFT validator.
- Check the SWIFT Error Codes list to understand the specific issue.
- Consult the institution's internal message template for compliance.
- Resubmit the corrected message.

Best Practices for MT 195

Baca Juga (Artikel Terkait)

- **automaker mini software version v8 erptoboss**
  URL: http://apexgalvanizing.com/automaker-mini-software-version-v8-erptoboss.pdf
- **automata and computability**
  URL: http://apexgalvanizing.com/automata-and-computability.pdf
- **automata and languages theory and applications**
  URL: http://apexgalvanizing.com/automata-and-languages-theory-and-applications.pdf
- **automata and mechanical toys**
  URL: http://apexgalvanizing.com/automata-and-mechanical-toys.pdf

Tags: #swift #message #example