Smart Bond Contract

Disintermediating the Life Cycle of a Tokenized Security

Pilot Report · March 2026

DZ BANK KFW

1	5	6	9	40
Public Blockchain used	Smart Contracts deployed	Market Participants connected	Functional States processed	Minutes until settled

Executive Summary

In March 2026 DZ BANK and KfW successfully piloted the usage of the Smart Bond Contract (SBC) protocol in a legally binding primary market transaction in form of a crypto security[1] (ISIN: DE000SBC0DZ4). The SBC is a smart-contract-based life cycle protocol for a tokenized security, with the pilot transaction demonstrating its complete deterministic operation on a public blockchain: From on-chain publication of funding levels and digital documentation including on-chain ISIN assignment, through automated registration in a crypto securities register, token minting, followed by decentralized Delivery-versus-Payment for issuance and final redemption settlement, all steps were executed deterministically by interconnected smart contracts on Polygon PoS[2]. The protocol connects the following six market participants in direct peer-to-peer interaction, with no central operator holding state: DZ BANK as Issuer, KfW as investor, WM Datenservice as ISIN provider, Cashlink GmbH as crypto securities registrar, Bundesbank as provider of the Trigger Solution enabling final cash settlement in central bank money in T2[3] plus a decryption oracle service hosted by NTT DATA.

Key Outcomes

- Smart Contract based securities' life cycle
- Functional workflow from indication to redemption
- Issuance settlement completed in under 40 minutes
- Peer-to-peer interaction; no intermediary service
- Digital documentation process (ICMA-BDT)
- Security identifier (ISIN) obtained automatically
- Open, modular, infrastructure-agnostic protocol

Rationale

As described in the whitepaper [1] the Smart Bond Contract concept is not tied to a platform — it is designed as infrastructure-agnostic protocol. Each process step can be executed deterministically on a selected blockchain with no central operator holding state or controlling the workflow. The functional workflow is transparent, modular, and follows clean design principles. The pilot proves that a complete deterministic issuance workflow of an electronic (tokenized) securities life cycle can be achieved with intraday finality and without introducing any platform dependency.

Contents

1 Within the meaning of §4(3) eWpG, German Electronic Securities Act

2 PoS: Proof of Stake

3 T2: Real-time Gross Settlement (RTGS) system operated by the Eurosystem

1 Summary

In March 2026 DZ BANK and KfW successfully piloted the usage of a Smart Bond Contract (SBC) [1] on Polygon PoS — a smart contract-based implementation of a complete life cycle of a tokenized security. The pilot transaction holistically covered a private placement's operational processes: From on-chain publication of funding levels and digital documentation including on-chain ISIN assignment, through automated registration in a crypto securities register, token minting, followed by decentralized ERC-7573[4] [2] Delivery-versus-Payment settlement, and ultimately extending to automated coupon payment and final redemption, all steps were executed deterministically and were mostly automized by interconnected smart contracts which were deployed in major parts directly within the life cycle on Polygon. At the core a smart contract (the SBC) acted as a digital intermediary responsible to orchestrate the life cycle and connecting adjoined services such as the token registrar functionality and ISIN assignment. The smart contracts enabled peer-to-peer interaction on the chosen DLT network — connecting the issuer, investor, ISIN provider, crypto registrar, and decryption oracle service — while the interoperability mechanism of Bundesbank Trigger Solution allowed for a key-based atomic Delivery-versus-Payment (DvP), linking asset token transfer to the release of previously reserved cash (cash lock).

> *"Our vision was to create an infrastructure-agnostic functional protocol design for the life cycle of tokenized security to prepare for a maximum possible level of automatism and operational risk reduction. For this initial pilot our choice was to use a public blockchain, as it offers the most native way to operate in a disintermediated way - A Smart Contract coordinated the interaction of the involved parties."*
>
> **Peter Kohl-Landgraf** · DZ BANK

Pilot Results

- Holistic frictionless tokenized securities life cycle from funding level publication, issuance to redemption
- T+0 settlement in T2 via Bundesbank Trigger Solution,
- Direct peer-to-peer interaction, no central orchestrator
- Standardised digital data object as single source of truth
- Settlement guaranteed through irrevocable cash-locking
- Open, modular, infrastructure-agnostic protocol
- Decentralized interoperability oracle mechanism

> *"The innovative mechanism for locking the payment and delivery of the asset was decisive. Once we locked our payment on the Bundesbank infrastructure and confirmed it via the Smart Bond interface, the delivery of the asset took place automatically within a few seconds."*
>
> **Dimitri Kunz** · KfW

2 Smart Contract Design

The following five contracts constructed a modular functional ecosystem responsible to orchestrate and manage the disintermediated processing of the transaction.

Contracts Deployed

- Smart Bond Factory — registry & access control
- Smart Bond Contract — life cycle state machine
- ERC-20 Token Registry — asset token ledger
- Smart ISIN Contract — security identifier assignment
- Decryption Oracle Contract — secure and stateless DvP

Smart Contracts as digital Intermediaries

Each transaction is served by a dedicated smart contract instance. A Smart Bond Contract instance enforces the agreed life cycle deterministically through a configurable state machine — coordinating all parties and guaranteeing settlement. These digital intermediary services are hosted as auditable distributed software; a central orchestrating agent is not needed. All interactions are transparent and independently verifiable on-chain.

Smart Bond Factory

The Smart Bond Factory contract enables the deployment and registration of a dedicated Smart Bond Contract instance for each transaction request and authorizes entitled parties to interact on specific predefined functionality — such as issuers publishing terms and funding levels, and investors submitting trade requests. In addition, it supports the cancellation of outstanding Smart Bond Contract instances in response to changed market conditions or other predefined cancellation triggers, in accordance with the applicable terms and conditions.

Smart Bond Contract

A Smart Bond Contract is a deterministic state machine orchestrating all transaction steps of the life cycle., holding digital issuance terms, obtaining an approved ERC-20 [3] allowance to perform the asset token transfer at settlement and requesting keys for processing Delivery-versus-Payment.

Smart ISIN Contract

The Smart ISIN Contract is a fully automated ISIN application process which functionally provides identifiers on-chain. Identifiers being stored on-chain in the smart bond digital terms and handed over to the registrar's API to automatically start the minting process.

> *"With this pilot transaction we became the first National Numbering Agency worldwide to provide an ISIN for a digital asset on-chain. For the first time, instrument reference data and ownership transfer are governed by the same code, thereby removing another layer of intermediation."*
>
> **Duc Au** · WM Datenservice

4 ERC = Ethereum Request for Comments. Standardization process for smart contract interfaces on Ethereum-compatible blockchains

Token Registry Contract

The customized ERC-20 asset token contract is deployed by the Crypto Securities Registrar, which previously retrieved the automated mint request from the issuer and the address is stored in the Smart Bond Contract which afterwards retrieves a transfer allowance to transfer the tokens within the settlement process. In this setup the crypto securities registrar is responsible for tracking ownership of the security and replaces the notary service function usually provided by Central-Securities-Depositories.

Decryption Oracle Contract

The decryption oracle contract provides the functionality required for a secure and stateless interoperable delivery-versus-payment. Based on the ERC-7573 reference implementation, the decryption oracle generates encrypted settlement keys with corresponding hashes and decrypts a key upon admissible request. The protocol ensures that the SBC is the only one allowed to request decryption. Storing the encrypted keys, the SBC triggers conditional key release upon transfer success or failure.

> *"We were integrated into the SBC ecosystem solely through the Decryption Contract and fully replaceable without any contract redeployment. There was no need for us to have any visibility into trade economics nor counterparty identities - exactly the right design."*
>
> **Maja Schwarz** · NTT Data

3 Digital Issuance Terms

Digital Issuance Terms parameters are structured in accordance with the ICMA Bond Data Taxonomy (BDT) [4] — one of the leading data model standards for fixed-income instrument description. The BDT record serves as the single source of information at the centre of the SBC's life cycle process: A structured digital source from which other objects such as digital issuance documentations (terms and conditions) and downstream processes derive exclusively from this canonical record. Transaction parameters are finalised on-chain by both issuer and investor.

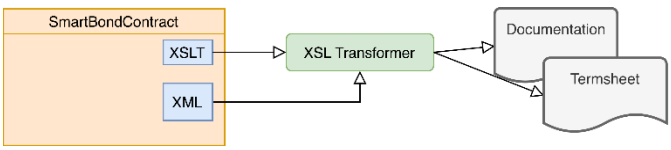

Figure 1: SmartBondContract as central source for documentation

ICMA BDT — DIGITAL DATA OBJECT

The BDT record specifies the digital issuance terms in XML[5] format. Legal terms are represented in an XSLT[6]. Both are stored in the Smart Bond Contract. An external XSL transformation engine enables export into any required downstream format. All from the same canonical on-chain source, with no manual intervention.

> *"The Smart Bond Contract revolutionizes legal bond documentation by combining ICMA BDT-standards and an automated deterministic issuance process with digital documentation using blockchain as a golden source for bond data. This ensures transparent, verifiable, and predictably certain execution throughout the tokenized securities lifecycle."*
>
> **Nick N. Wittek** · Jones Day

4 Deterministic Life Cycle Model

In alignment with the GFIF Guidelines of the Monetary Authority of Singapore [5] the Smart Bond Contract implements a deterministic life cycle in form of a configurable, modular state machine. Each state strictly gates the next: no participant can invoke a function unless the contract is in the required state and holds the required role. All involved parties — issuer, investor, ISIN provider, crypto registrar and decryption oracle — are connected into one automated, peer-to-peer sequence. No central platform coordinates interactions: The Smart Bond Contract itself acts as the digital intermediary. For the sake of the PoC, certain steps (e.g. creating a trade request) were executed manually via the dApp[7] frontend. In principle, however, the protocol supports fully automated A2A interaction for every step, reducing the need for manual intervention to a minimum.

Transaction Request & Contract Deployment

The issuer publishes its underlying programme, issuance terms and indicative funding levels on-chain via the Smart Bond Factory. When an eligible investor selects terms and creates a trade request on a respective quote the factory deploys a dedicated Smart Bond Contract instance initialised with the relevant indicative parameters. The digital term sheet is created and stored directly in the contract in ICMA BDT XML format. An XSLT-based XML-converter enables export into any downstream format and links it to the legal issuance terms and conditions, also provided in a digital format. The Life cycle status changes to "`IndicationPhase`".

Transaction Data Confirmation

In the following step by calling the "inceptTrade" function the issuer publishes finalised BDT parameters on-chain. By calling "confirmTrade", the investor verifies and submits a signed hash-based confirmation following ERC-6123 [6] for bilateral transaction data verification. Once confirmed, the contract advances to the next state with all issuance terms become immutable — no further modification is possible. The SBC state moves to "`DistributionPhase`".

Minting Process and Public Distribution

The Smart Bond Contract triggers an on-chain ISIN request to the Smart ISIN Contract. The ISIN provider's system monitors the contract, validates parameters, and writes the allocated

5 XML: A markup language for representing structured data in text form.

6 XSLT: A transformation language for converting XML into other text-based formats.

7 dApp = Distributed Application

identifier back on-chain. A contract event then triggers a fully automated mint request to the Crypto Registrar's off-chain API. The complete BDT-structured issuance terms serve as the standardized data reference base. The registrar deploys the ERC-20 token contract [3], mints the asset tokens and returns the token address. The Smart Bond Contract then obtains an approved ERC-20 allowance, enabling it to perform the atomic token transfer at settlement. Although triggered by a smart contract event the mint request in this pilot was primarily via off-chain API. Full on-chain interaction is targeted for the next iteration.

> *"The automated handover was seamless. We received the full instrument specification, minted the tokens, and returned the token address to the Smart Bond Contract — entirely without manual intervention. One of the most innovative tokenization projects we have seen on the market "*
>
> **Simon Censkowsky** · Cashlink GmbH

Interoperable Delivery-versus-Payment (DvP)

To perform an atomic DvP across the separated asset and cash infrastructures, the protocol made use of the ERC-7573 standard proposal [2], enabled through a decentralized oracle-based interoperability service. The Bundesbank Trigger Solution provides a service for the settlement of DLT-based wholesale financial transactions in central bank money in T2 accessible via A2A, providing the parties the possibility to lock a certain amount of cash against the presentation of a secret key that validates against a hash.

> *"The piloted use of the SBC protocol shows how tokenized assets and central bank money can interact securely across platforms in practice. The Trigger Solution enables atomic settlement, reduces counterparty risk, and opens a practical path towards interoperable tokenized markets without relying on a single infrastructure."*
>
> **Markus Zschocke** · Deutsche Bundesbank

The interoperable settlement mechanism used in this protocol works as follows: The Smart Bond Contract request an encrypted transfer key and the corresponding key hash from the Decryption Oracle Contract and stores both. The investor uses the key hash to lock the cash amount — eliminating settlement failure risk at this point. Once the Smart Bond Contract verifies the successful transfer of the asset tokens to the designated investor custody wallet address, it requests the decryption of the transfer key from the on-chain Decryption Oracle Contract to trigger automated cash release. The decryption oracle requires zero visibility into transaction details which enables loose coupling between the off-chain oracle service and the on-chain DvP protocol [7].

CASH LOCKING & STATELESS ORACLE

Cash is irrevocably locked before asset transfer — principal risk structurally eliminated. The oracle is fully stateless: it holds no transaction data, presents no attack surface, and can be replaced by authorized parties without any on-chain change.

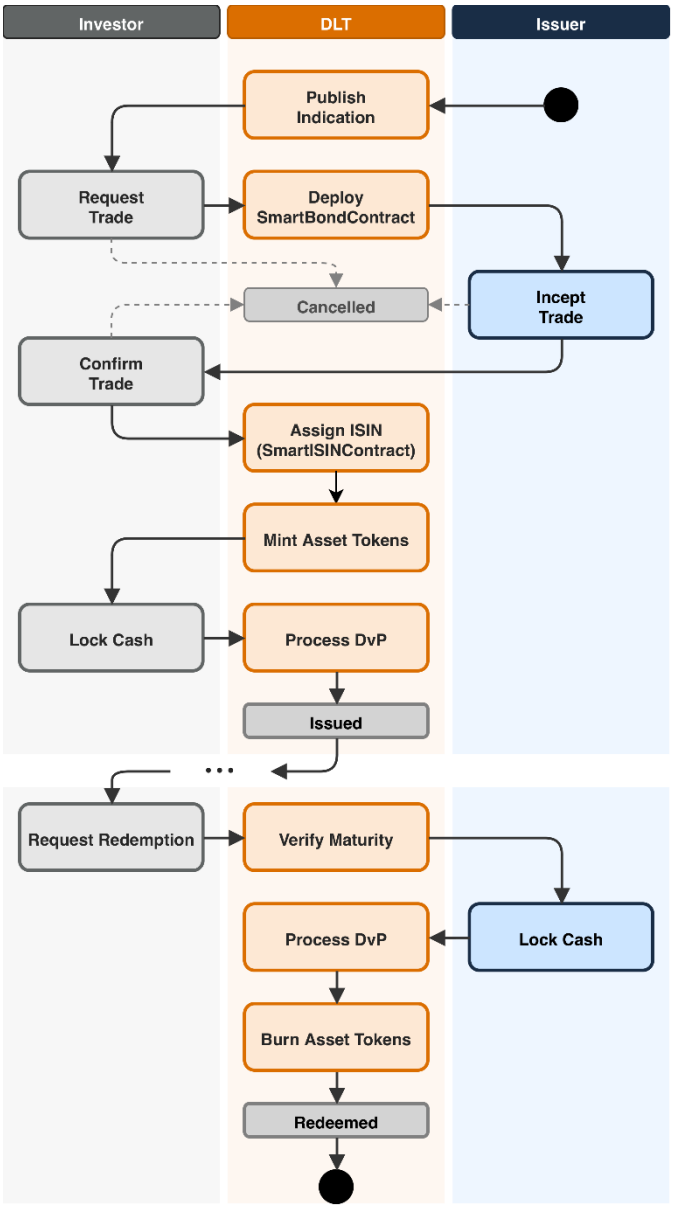

Figure 2: SBC Functional Sequence Flow

4.5 Coupon Payment & Redemption

On a scheduled coupon date or the final redemption date, the described DvP process runs in reverse: The issuer generates the encrypted-hashed-key pair and locks the cash on Trigger Solution (in the Pilot this was done automated via A2A), any party triggers the transfer via the Smart Bond Contract, and, if admissible, the oracle releases the transfer key, which is claimed by the investor. Upon redemption, returned tokens are permanently burned on-chain. The contract advances to 'Redeemed', completing the full life cycle.

5 Decentralised Application Architecture

The protocol is accompanied by a three-layer dApp — smart contract layer, backend service layer, and frontend application layer — each independently deployable. All authoritative state resides in the on-chain contracts. The backend handles event monitoring and external service integration; the frontend provides role-specific interfaces and a live on-chain event monitor. Any participant can run their own backend connecting to the same contracts.

SMART CONTRACT— SINGLE SOURCE OF TRUTH

On-chain contracts hold all authoritative state and operate autonomously regardless of backend or frontend availability. Any participant can verify any state transition by reading the contracts directly.

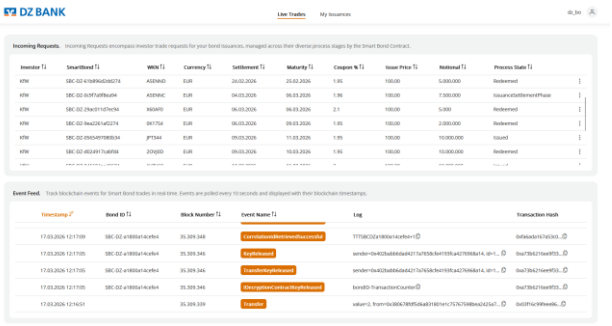

Figure 3: SBC Frontend for manual interaction with live Event Monitor

6 Infrastructure Choice & Privacy

The transaction was implemented and conducted on a public blockchain infrastructure as a deliberate choice to prioritize the holistic design of a full life cycle functional workflow before addressing privacy concerns in the second step.

Infrastructure-Agnostic by Design

The protocol is composable and infrastructure-agnostic. Polygon PoS was selected for this pilot for its throughput, low transaction cost, and EVM compatibility. The deterministic life cycle logic, however, does not depend on any specific blockchain. Using a public blockchain for that pilot transaction, full transparency was a deliberate choice: it enabled independent verifiability for all participants and regulators. The functional design applies across public and permissioned environments. Privacy might be addressed at the application layer via selective encryption and moving sensible information off-chain. The encrypted transfer key mechanism (ERC-7573) means that sensitive details were never exposed on-chain. This makes the Smart Bond Contract a candidate for standardisation as shared financial market infrastructure — open, composable, and usable across the capital markets ecosystem.

ARCHITECTURE: PROTOCOL, NOT PLATFORM

Each participant interacts against the digital smart contract eco system. The protocol is composable and applicable across different infrastructures. Since a public infrastructure is chosen no platform operator is needed to orchestrate the interaction between the involved participants

A modular wallet concept is used to ensure secure smart contract interaction and resilient digital asset custody by separating an operational interaction wallet from a dedicated custody wallet.

7 Pilot Results

The following section summarizes the pilot results. In the pilot transaction configuration, several steps were deliberately kept manual; however, the modular protocol design already prepares these steps for further automation. The setup is agnostic with respect to the underlying infrastructure and flexible in deciding which process steps are executed off-chain versus on-chain.

Settlement 40 min · T+0 same-day · Peer-to-peer

- Full issuance life cycle — transaction initiation to settlement — under 45 minutes
- Direct peer-to-peer interaction across all six parties; no central intermediary
- Smart Bond Factory: creates one dedicated contract instance per transaction
- ICMA BDT as central digital data object — XML/XSLT issuance terms and conditions with various HTML downstream formats
- Registrar integration fully automated; full on-chain interaction in next iteration
- Conditional-upon-transfer DvP: Asset chain and payment system linked via ERC-7573, interaction on Trigger Solution via A2A and U2A
- Cash locking: irrevocable settlement guarantee — principal risk eliminated
- ISIN allocated on-chain with zero manual intervention
- DvP Oracle: zero knowledge of transaction economics or counterparty identities
- All life cycle events publicly observable and independently auditable on-chain

8 The Case for Open Protocols

We publish this protocol in a highly transparent manner to contribute to open, interoperable standards for tokenized capital market transactions. The critical distinction is architectural: the life cycle is an openly documented, modular set of auditable functions — not a closed software solution tied to a platform operator.

No Platform. No Operator. Only Protocol.

In traditional tokenized finance, a platform operator stands between participants — routing messages, holding state, and controlling access. The Smart Bond Contract removes this dependency architecturally. Each participant integrates directly against the open smart contract interface. No operator can intervene in, delay, or selectively deny access to the life cycle. No single entity sets prices or can unilaterally change the rules. The public blockchain provides the shared execution environment: neutral, permissionless at the infrastructure level, and independently verifiable by all participants and regulators.

Open Design for further Development

Looking beyond the MVP[8] version used for the PoC, the smart contract code, the BDT data model, and the dApp architecture can be made available for review and, where appropriate, integration. DZ BANK AG is open to dialogue with issuers, investors, registrars, infrastructure operators and regulators to discuss the design and its possible applications and enhancements. An open, infrastructure-agnostic protocol standard for tokenized bond issuance will ultimately depend on broad industry participation.

This pilot transaction serves as a starting point, not a finished product.

9 References and Contacts

References

[1] DZ BANK AG, "Smart Bond Contract," 2025. https://www.dzbank.de/content/dam/dzbank/dokumente/en/dz-bank/Press/press_releases/2025/DZ BANK_Whitepaper Smart Bond Contract.pdf.

[2] C. P. Fries and P. Kohl-Landgraf, “ERC-7573: Conditional-upon-Transfer-Decryption for DvP,” 2023. http://eips.ethereum.org/EIPS/eip-7573.

[3] F. Vogelsteller and V. Buterin, “ERC-20: Token Standard,” Ethereum Improvement Proposals, https://eips.ethereum.org/EIPS/eip-20.

[4] ICMA, “Bond Data Taxonomy,” https://www.icmagroup.org/fintech-and-digitalisation/fintech-advisory-committee-and-related-groups/bond-data-taxonomy/.

[5] Monetary Authority of Singapore, “Guardian Fixed Income Framework,” 2024. https://www.mas.gov.sg/schemes-and-initiatives/project-guardian.

[6] C. P. Fries, P. Kohl-Landgraf and A. Korpis, “ERC-6123: Smart Derivative Contract: A deterministic protocol for frictionless trade processing of financial contracts,” 2022. http://eips.ethereum.org/EIPS/eip-6123.

[7] C. P. Fries and P. Kohl-Landgraf, “Stateless and Secure Delivery versus Payment across Blockchains,” 2023. http://ssrn.com/abstract=4628811.

Contacts

DZ BANK

Henning Ahmann (Capital Markets Institutional Clients)
Annina Boehnke (Client Relationship Management)
Nico Braun (Fixed Income Syndicate)
Julian Eicke (Security Services & Digital Custody)
Christian Fries (Group Risk Control)
Thorsten Hirsch (IT Department)
Martin Kötter (Capital Markets Trading)
Peter Kohl-Landgraf (Group Treasury)
Raphael Prandtl (Group Risk Control)
Frederick Spahn (Group Treasury)

KFW

Stefanie Barnewald (Transaction Management)
Valerio Jacobi (Financial Markets)
Dimitri Kunz (Financial Markets)

8 MVP = Minimum Viable Product