

SUBSEMBLY

Banking Apps & APIs

BankAccessServer

Produktinformation

Version 3.5.5

Subsembly GmbH

Hofmannstr. 7b
81379 München

<http://subsembly.com>

bas@subsembly.com

Stand: 14.11.2024

Inhaltsverzeichnis

Überblick	3
Für Drittanbieter nach PSD2 (TPP)	3
Für Unternehmen	4
Bankenprotokolle und Standards	4
NextGenPSD2	4
FinTS	5
XS2A (Scraper/native APIs)	5
EBICS	5
Unterstützte Banken	5
Deployment	6
Deployment-Optionen	6
Geeignet für gängige Cloud-Plattformen	6
Containerisierung mit Docker	6
Logging und Monitoring	7
Integration in bestehende Umgebungen	7
Development	8
Einheitliche REST API für verschiedene Bankprotokolle/Standards	8
OpenAPI/Swagger Unterstützung	8
Plattformunabhängig und in alle Anwendungsszenarien / Programmiersprachen integrierbar	8
Umfangreiche Dokumentation und Beispiele	8
Testserver für FinTS/EBICS für einen schnellen Start	8
Praxiserprobt und bei vielen TPPs/Unternehmen im täglichen Einsatz	9
Lizenzierung	9
Weitere Informationen	10
Subsembly BankAccessServer	10
Subsembly Banking APIs / SDKs	10
Spezifikationen	10
NextGenPSD2 Access to Account Interoperability Framework	10
Laufzeitumgebung	10
Codegenerierung	10

Überblick

Der **Subsembly BankAccessServer (BAS)** ist eine leistungsstarke und flexible Plattform, die den sicheren Zugriff auf Bankkontodaten und die Ausführung von Zahlungsaufträgen ermöglicht. Mit der Unterstützung für **FinTS**, **EBICS**, nativen APIs/Scrapern und der **PSD2 API der Berlin Group** bietet der BAS eine einheitliche **REST API**, die einfach in jede Umgebung integriert werden kann.

Praxiserprobt und bei vielen Banken, Drittanbietern und Unternehmen im täglichen Einsatz, bietet der BAS plattformunabhängige Unterstützung und flexible Deployment-Optionen sowohl On-Premise als auch in der Cloud.

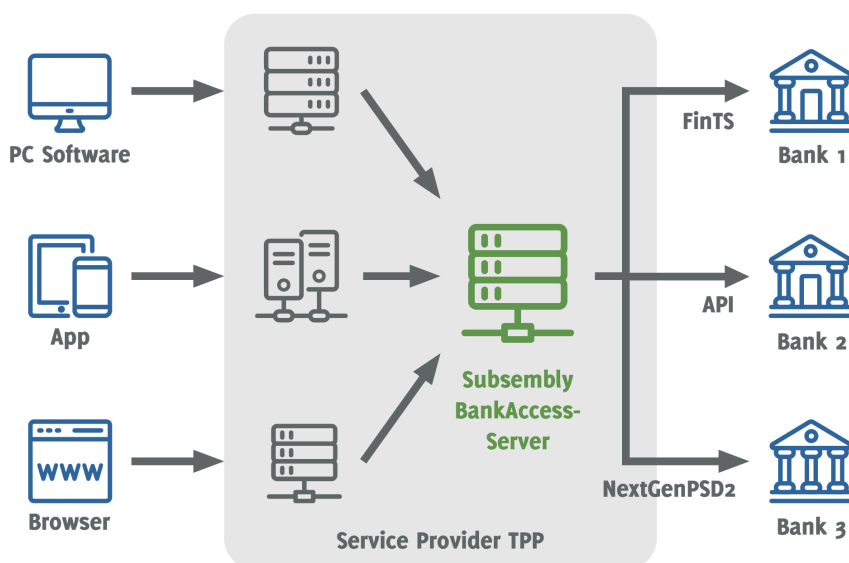
Der BAS ermöglicht die Erstellung kostengünstiger, skalierbarer und flexibler Lösungen für den multibankenfähigen Kontenzugriff, ohne dass hierfür ein externer Dienstleister zwischengeschaltet werden muss.

Für Drittanbieter nach PSD2 (TPP)

Der BankAccessServer wird als zentrale Komponente für die Kommunikation mit den Banken über verschiedene Zugangswege, wie FinTS, NextGenPSD2, proprietäre APIs oder Screen-Scraper, eingesetzt.

Kontoinformationsdienste (AISP) - z.B. Kontowechselservice, Cloud-Faktura/Buchhaltung, Multi-Banking Apps, Scoring, Mandatserteilung

Zahlungsauslösedienste (PISP) - z.B. Online-/E-Commerce Zahlungen, P2P Payments, Multi-Banking Apps mit Bezahlungsfunktion

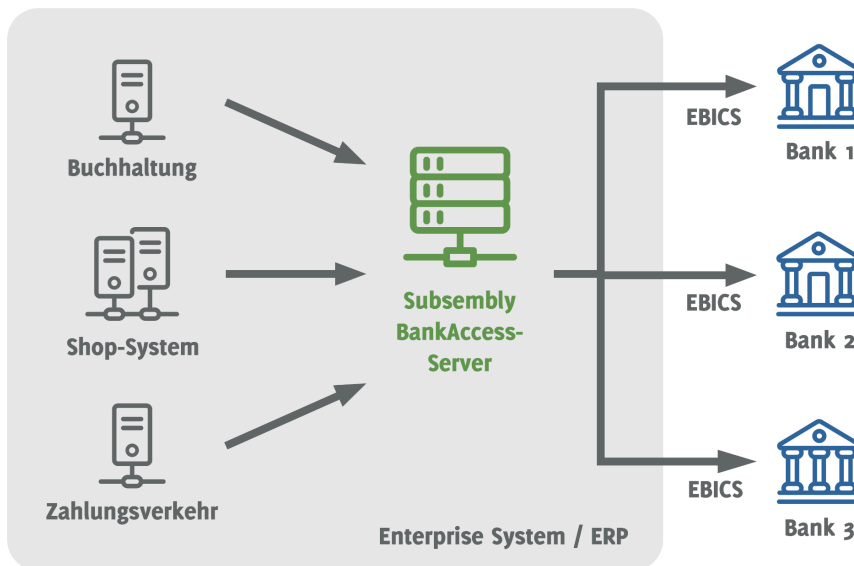


Für Unternehmen

Der BankAccessServer verbindet Ihre Unternehmenssoftware direkt mit Ihren Bankkonten, ohne Zwischenschaltung eines weiteren (cloudbasierten) Serviceanbieters. Für diese Lösung ist keine BaFin-Erlaubnispflicht nach ZAG (PSD2) erforderlich.

Gewerblicher Zahlungsverkehr via EBICS - z.B. Einzel- und Sammellastschriften / Einzel- und Sammelüberweisungen

Automatisierung der Buchhaltung - z.B. Abgleich offener Posten, Umsatzkategorisierung, Liquiditätsplanung



Bankenprotokolle und Standards

Der BankAccessServer verbindet sich durch einzeln lizenzierbare Konnektoren mit den verschiedenen Schnittstellen der Bankensysteme. Je nach Anwendungsfall kommen andere Konnektoren zum Einsatz.

NextGenPSD2

Von der BaFin lizenzierte Third-Party-Provider (TPP) können über eine PSD2 Schnittstelle nach Berlin Group Standard auf die Zahlungsverkehrskonten ihrer Kunden zugreifen. Kontoumsätze und Kontosalde stehen Account Information Service Providern (AISP) zur Verfügung. Um Zahlungen auszulösen, ist eine zusätzliche BaFin Lizenzierung als Payment Initiation Service Provider (PISP) erforderlich. Eine Liste aller über PSD2 erreichbaren Banken finden Sie [hier](#).

FinTS

Kontoarten, die nicht durch die PSD2 Schnittstelle abgedeckt werden, z.B. Anlagen, Termingelder oder Wertpapierdepots, können häufig über die FinTS 3.0 Standardschnittstelle abgerufen werden, welche von den meisten Banken in Deutschland angeboten wird. Gemäß ZAG (Zahlungsdiensteaufsichtsgesetz) darf über diese Schnittstelle nicht auf Zahlungsverkehrskonten zugegriffen werden. Eine Liste aller über FinTS erreichbaren Banken finden Sie [hier](#).

XS2A (Scraper/native APIs)

Kontoarten, die weder durch die PSD2 Schnittstelle noch durch FinTS abgedeckt werden, können in vielen Fällen über Web-Scraping ausgelesen werden. Häufig ist dies bei Kreditkarten der Fall. Eine Aufstellung aller Banken und Kreditkartenanbieter, welche über Web-Scraping abgerufen werden können, finden Sie [hier](#).

EBICS

Für die Integration des geschäftlichen Zahlungsverkehrs empfiehlt sich der Kontozugriff über die EBICS Schnittstelle, welche von Banken in Deutschland, Österreich und der Schweiz angeboten wird. Da diese Schnittstelle nur für gewerbliche Kunden verfügbar ist, ist keine BaFin-Erlaubnispflicht nach ZAG (PSD2) erforderlich.

Unterstützte Banken

Die Liste der unterstützten Banken finden Sie unter folgenden URLs:

Protokollstandard	Bankenliste
PSD2	https://subsembly.com/berlingroup.html
FinTS	https://subsembly.com/banken.html
Scraper/native APIs	https://subsembly.com/screen-scraper.html

Deployment

Deployment-Optionen

Der BankAccessServer bietet flexible Deployment-Optionen, die sowohl On-Premise als auch Cloud-basierte Installationen unterstützen.

On-Premise Deployment: Für Unternehmen, die ihre Server-Infrastruktur selbst verwalten möchten, bietet der BankAccessServer eine ideale Lösung zur Installation und zum Betrieb in unternehmenseigenen Rechenzentren. Diese Option ermöglicht eine vollständige Kontrolle über die Infrastruktur und stellt sicher, dass alle internen Sicherheits- und Compliance-Richtlinien eingehalten werden.

Cloud Deployment: Der BankAccessServer kann auch in Cloud-Umgebungen bereitgestellt werden, was Flexibilität und Skalierbarkeit bietet. Er unterstützt gängige Cloud-Plattformen wie Microsoft Azure und Amazon Web Services (AWS). Dies ermöglicht die Nutzung von Cloud-spezifischen Diensten und Features, die den Betrieb und die Verwaltung erleichtern.

Geeignet für gängige Cloud-Plattformen

Der BankAccessServer kann mit verschiedensten Cloud-Plattformen verknüpft werden. Beispielfhaft sind hier Microsoft Azure oder Amazon Web Services genannt, es sind aber auch andere Cloud-Lösungen möglich.

Microsoft Azure: Der BankAccessServer lässt sich nahtlos in Azure-Dienste integrieren, wie zum Beispiel Azure Key Vault für die sichere Verwaltung von Geheimnissen und Azure Monitor für umfassendes Monitoring. Die Bereitstellung kann als Azure App Service oder in Kubernetes-Clustern erfolgen, was eine einfache Skalierung und Verwaltung ermöglicht.

Amazon Web Services (AWS): Auf AWS kann der BankAccessServer von den Diensten AWS Secrets Manager zur sicheren Verwaltung von Geheimnissen und AWS CloudWatch für umfassendes Monitoring profitieren. Die Bereitstellung kann in Amazon ECS, EKS oder als EC2-Instanzen erfolgen, was eine flexible und skalierbare Lösung bietet.

Containerisierung mit Docker

Der BankAccessServer unterstützt Docker und bietet vorgefertigte Docker-Images, die eine einfache und schnelle Bereitstellung ermöglichen. Durch die Unterstützung von Container-Orchestrierungssystemen wie Kubernetes können Unternehmen Microservices effizient verwalten und skalieren.

Logging und Monitoring

Der BankAccessServer kann mit verschiedensten Logging- und Monitoring-Tools analysiert und überwacht werden.

Logging: Der BankAccessServer integriert sich nahtlos mit dem Elastic Stack (ELK), was eine umfassende Log-Analyse ermöglicht. Dies unterstützt zentralisiertes Logging und Monitoring, wodurch Unternehmen die Leistung und Sicherheit ihrer Anwendungen in Echtzeit überwachen können.

Elastic Cloud: Für skalierbares Logging und Monitoring kann Elastic Cloud genutzt werden. Dies ermöglicht die Analyse und Visualisierung von Logs in Echtzeit, was eine schnelle Fehlerbehebung und Optimierung ermöglicht.

Integration in bestehende Umgebungen

Ein weiterer Vorteil des BankAccessServers ist, dass keine zusätzlichen Datenbanken notwendig sind. Er kann nahtlos in bestehende IT-Umgebungen und Datenbanken integriert werden, was die Nutzung vorhandener Infrastruktur für Speicherung und Verarbeitung ermöglicht. Dies reduziert die Komplexität und Kosten der Implementierung.

Development

Einheitliche REST API für verschiedene Bankprotokolle/Standards

Der BankAccessServer bietet eine einheitliche REST API, die verschiedene Bankprotokolle und Standards unterstützt, darunter FinTS (HBCI), EBICS, native APIs/Scraper und die PSD2 API der Berlin Group. Die REST API ermöglicht es Entwicklern, Finanzanwendungen effizient und flexibel zu entwickeln und dabei nahtlos zwischen den verschiedenen Protokollen zu agieren.

OpenAPI/Swagger Unterstützung

Durch die Unterstützung des OpenAPI-Standards und die Integration von Swagger können Entwickler von einer automatisierten und umfassenden API-Dokumentation profitieren. Swagger UI ermöglicht interaktive API-Tests direkt im Browser, während OpenAPI-Spezifikationen die Erstellung von Client-Bibliotheken und SDKs in verschiedenen Programmiersprachen erleichtern.

Plattformunabhängig und in alle Anwendungsszenarien / Programmiersprachen integrierbar

Der BankAccessServer ist plattformunabhängig und lässt sich in eine Vielzahl von Anwendungsszenarien und Programmiersprachen integrieren. Dank der auf .NET Core basierenden Architektur ist er mit Windows, Linux und macOS kompatibel. Die REST API kann von jeder Programmiersprache aus genutzt werden, die HTTP-Anfragen unterstützt, einschließlich Python, Java, C#, JavaScript und mehr.

Umfangreiche Dokumentation und Beispiele

Um Entwicklern den Einstieg zu erleichtern, bietet der BankAccessServer eine umfangreiche Dokumentation und zahlreiche Beispiele. Detaillierte API-Referenzen, Schritt-für-Schritt-Anleitungen und praxisnahe Code-Beispiele helfen dabei, die API schnell und effizient zu integrieren und zu nutzen.

Testserver für FinTS/EBICS für einen schnellen Start

Für einen schnellen und unkomplizierten Start stellt der BankAccessServer Testserver für FinTS und EBICS bereit. Diese Testumgebungen ermöglichen es Entwicklern, ohne den Einsatz von Echtdateien oder echten Bankkonten zu arbeiten, und bieten eine sichere Umgebung zum Testen und Experimentieren.

Praxiserprobt und bei vielen TPPs/Unternehmen im täglichen Einsatz

Der BankAccessServer ist praxiserprobt und wird täglich von zahlreichen Drittanbietern (TPPs) und Unternehmen eingesetzt. Diese weitreichende Nutzung belegt die Zuverlässigkeit und Leistungsfähigkeit der Plattform, die sich in verschiedensten Anwendungsfällen bewährt hat.

Lizenzierung

Die Lizenz gibt es in verschiedenen Preisstufen in Abhängigkeit von Nutzerzahlen und Unternehmensgröße. Die Lizenzierung ist unabhängig von der Anzahl der abgewickelten Transaktionen.

Die einzelnen Bankschnittstellen FinTS/HBCI, EBICS und XS2A/PSD2 können unabhängig lizenziert werden.

Zusätzlich zur Lizenzierung ist der Abschluss eines Wartungs- und Supportvertrages obligatorisch. Darüber hinaus bieten wir Vor-Ort-Schulungen und Projektunterstützung.

Weitere Informationen

Subsembly BankAccessServer

Webseite: <https://subsembly.com/bank-access-server.html>

OpenAPI Spezifikation: <https://subsembly.com/apidoc/bas/>

Produktinformationen: <https://subsembly.com/download/BankAccessServer.pdf>

API Spezifikation: <https://subsembly.com/download/BankAccessServerClientInterface.pdf>

Installation: <https://subsembly.com/download/BankAccessServerInstallation.pdf>

EBICS: <https://subsembly.com/download/BankAccessServerEBICS.pdf>

Subsembly Banking APIs / SDKs

FinTS API: <https://subsembly.com/fints-api.html>

Ebics API: <https://subsembly.com/ebics-api.html>

SEPA API: <https://subsembly.com/xs2a-api.html>

XS2A API: <https://subsembly.com/xs2a-api.html>

Spezifikationen

Subsembly Payments Datenformate (SUPA): <https://subsembly.com/supa.html>

Deutsche Kreditwirtschaft / EBICS:

<https://die-dk.de/zahlungsverkehr/electronic-banking/dfu-verfahren-ebics/>

Deutsche Kreditwirtschaft / FinTS:

<https://die-dk.de/zahlungsverkehr/electronic-banking/fints/>

Deutsche Kreditwirtschaft / PSD2 Kontoschnittstelle:

<https://die-dk.de/zahlungsverkehr/electronic-banking/psd2-kontoschnittstelle/>

NextGenPSD2 Access to Account Interoperability Framework

PSD2 Access to Bank Accounts

<https://www.berlin-group.org/psd2-access-to-bank-accounts>

Laufzeitumgebung

Microsoft .NET Core: <https://dotnet.microsoft.com/en-us/download/dotnet>

Codegenerierung

Swagger CodeGen: <http://swagger.io/swagger-codegen/>