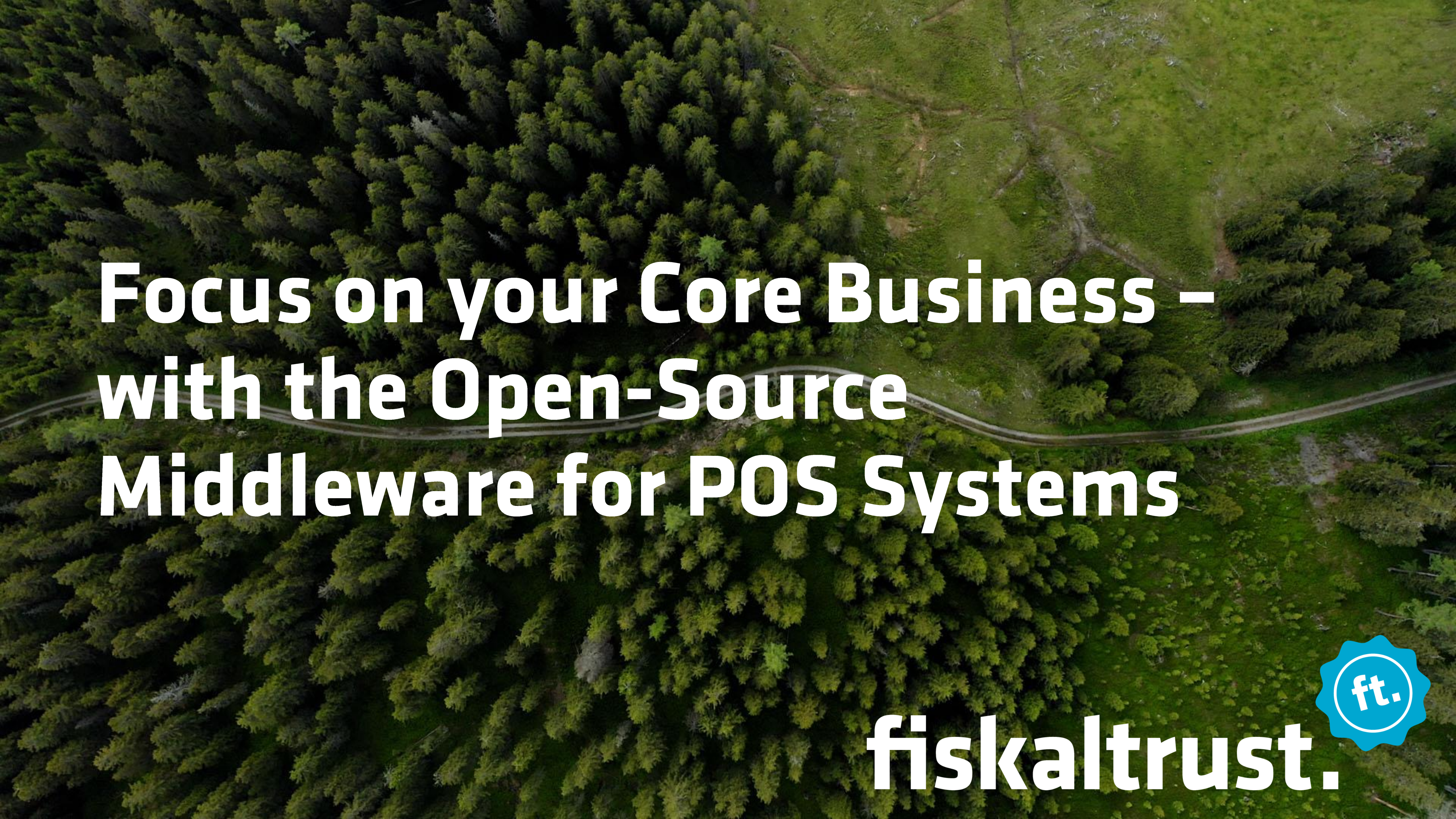


Focus on your Core Business – with the Open-Source Middleware for POS Systems

fiskaltrust.

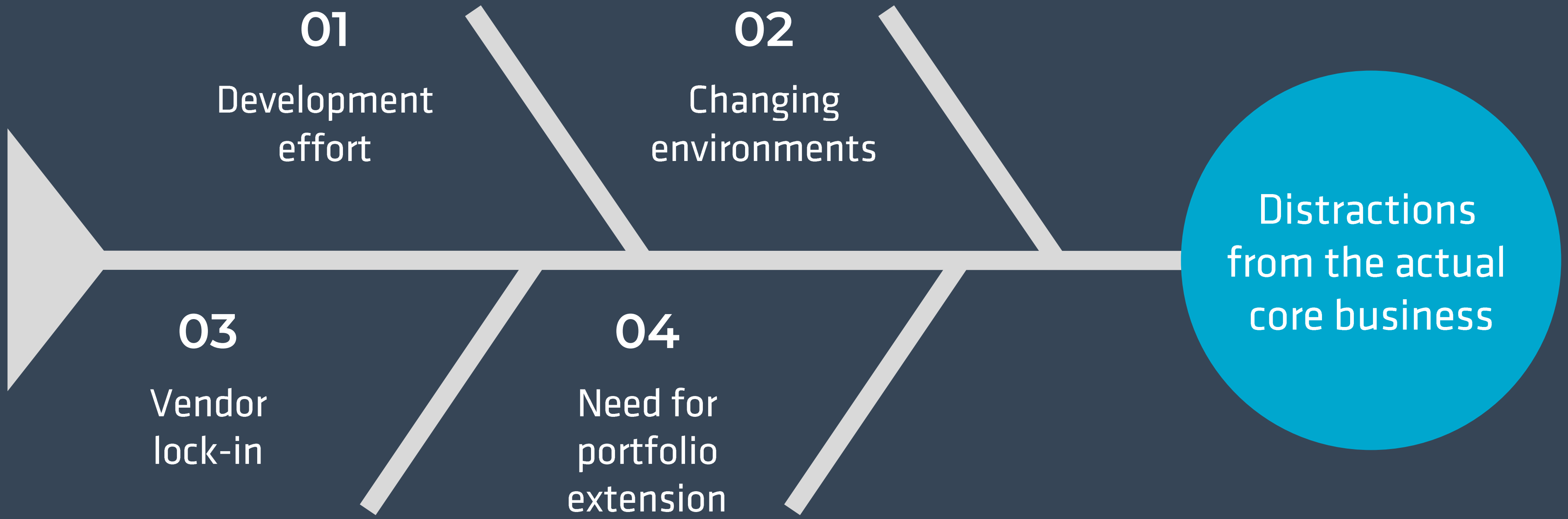
We know how to deal with local regulations, payments, value-added-taxes, and receipts.

Marco Leidl
VP Customer Success
linkedin.com/in/marcoleidl

Tom Schmiedlechner
COO | VP Open Source & DevRel
linkedin.com/in/tschmiedlechner

fiskaltrust.

Challenges when implementing POS software



01

Development effort

02

Changing environments

03

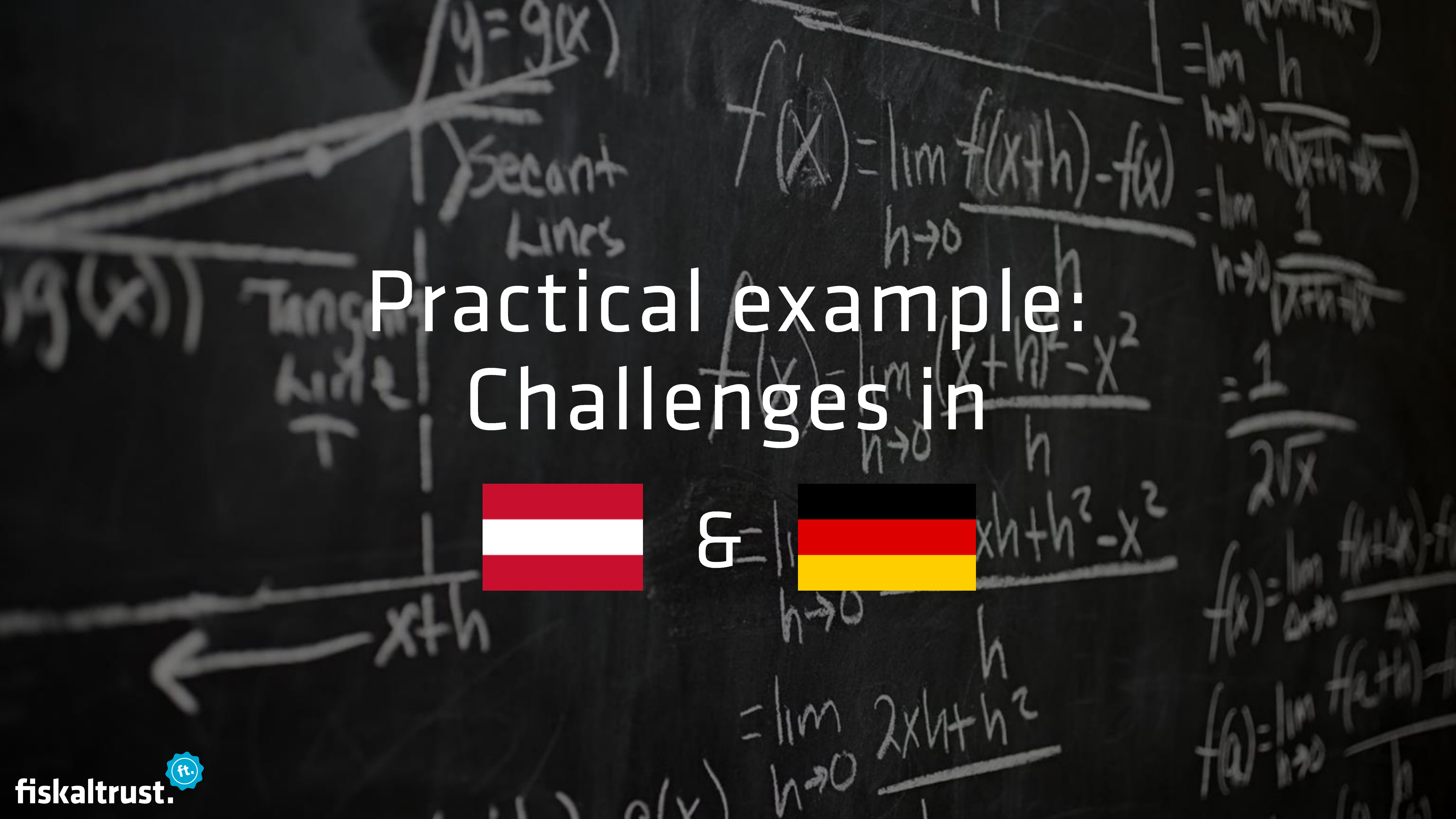
Vendor lock-in

04

Need for portfolio extension

Distractions from the actual core business

fiskaltrust.



Practical example: Challenges in

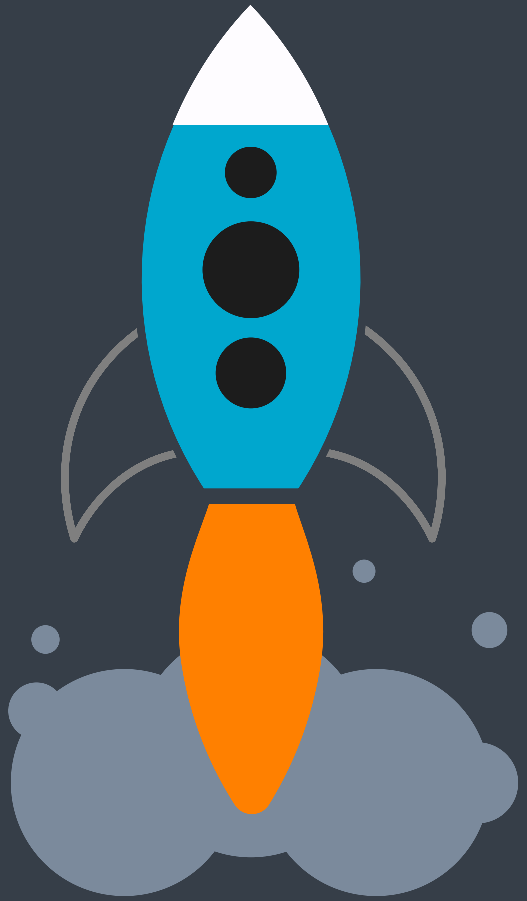
🇦🇹 & 🇩🇪

What the focus should be

Build products and features for your industry, your key partners, and your markets

Deliver the experience your users love

Add value with digital receipts, payment and more

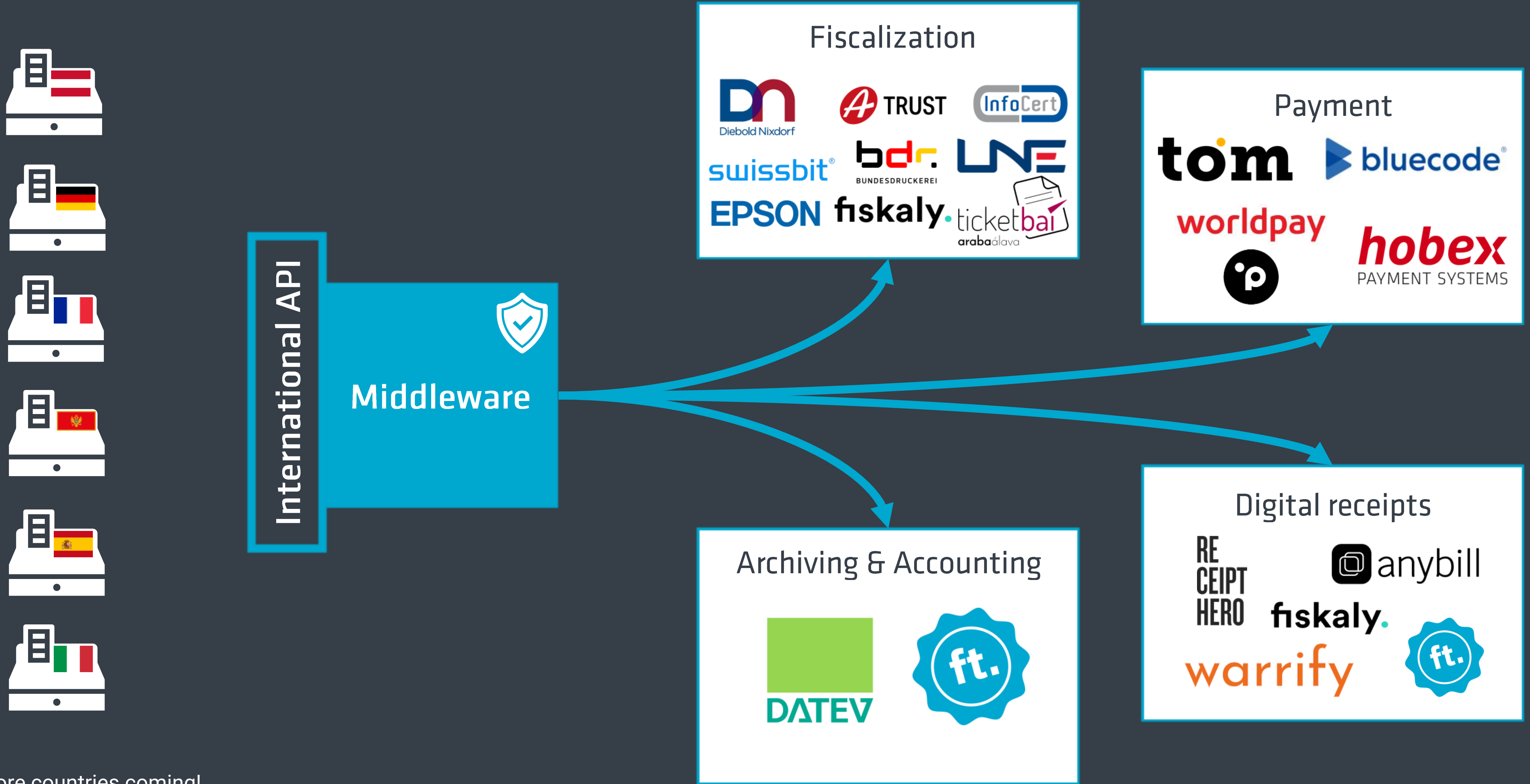


Build flexible solutions your vendors can compose individually

Create products that stand out from the competition

fiskaltrust.

The fiskaltrust.Middleware



International API

Middleware

Fiscalization

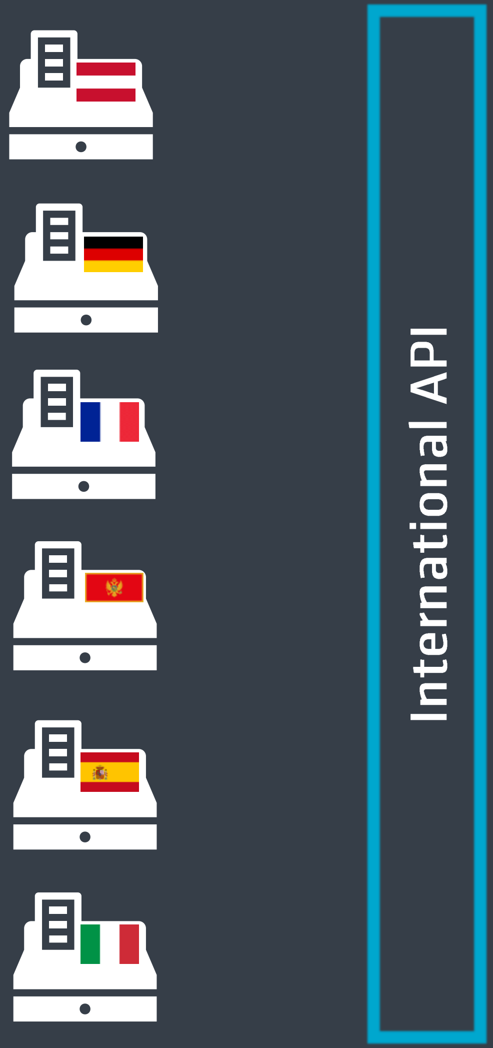
Payment

Archiving & Accounting

Digital receipts

… with more countries coming!

The international POS API

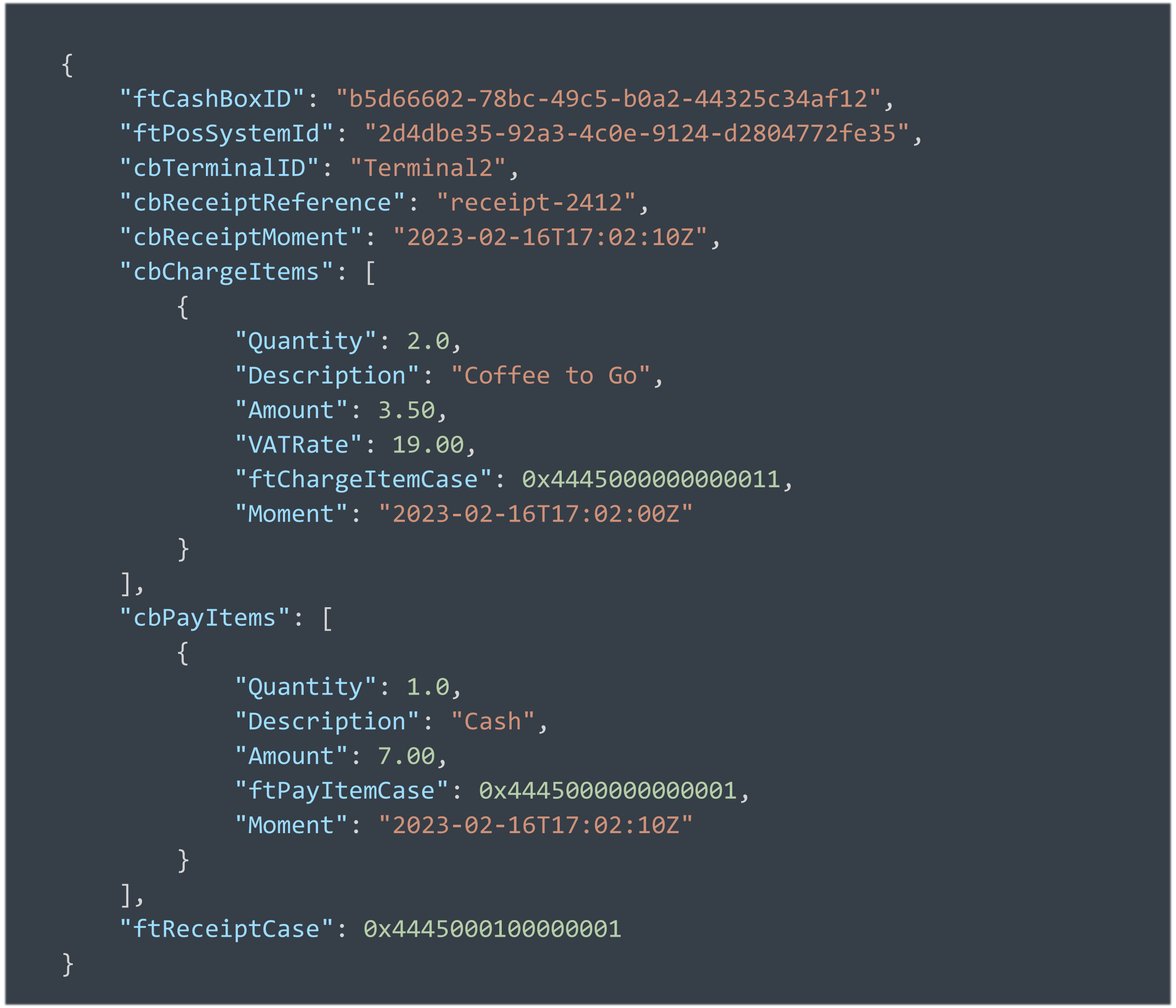


International API

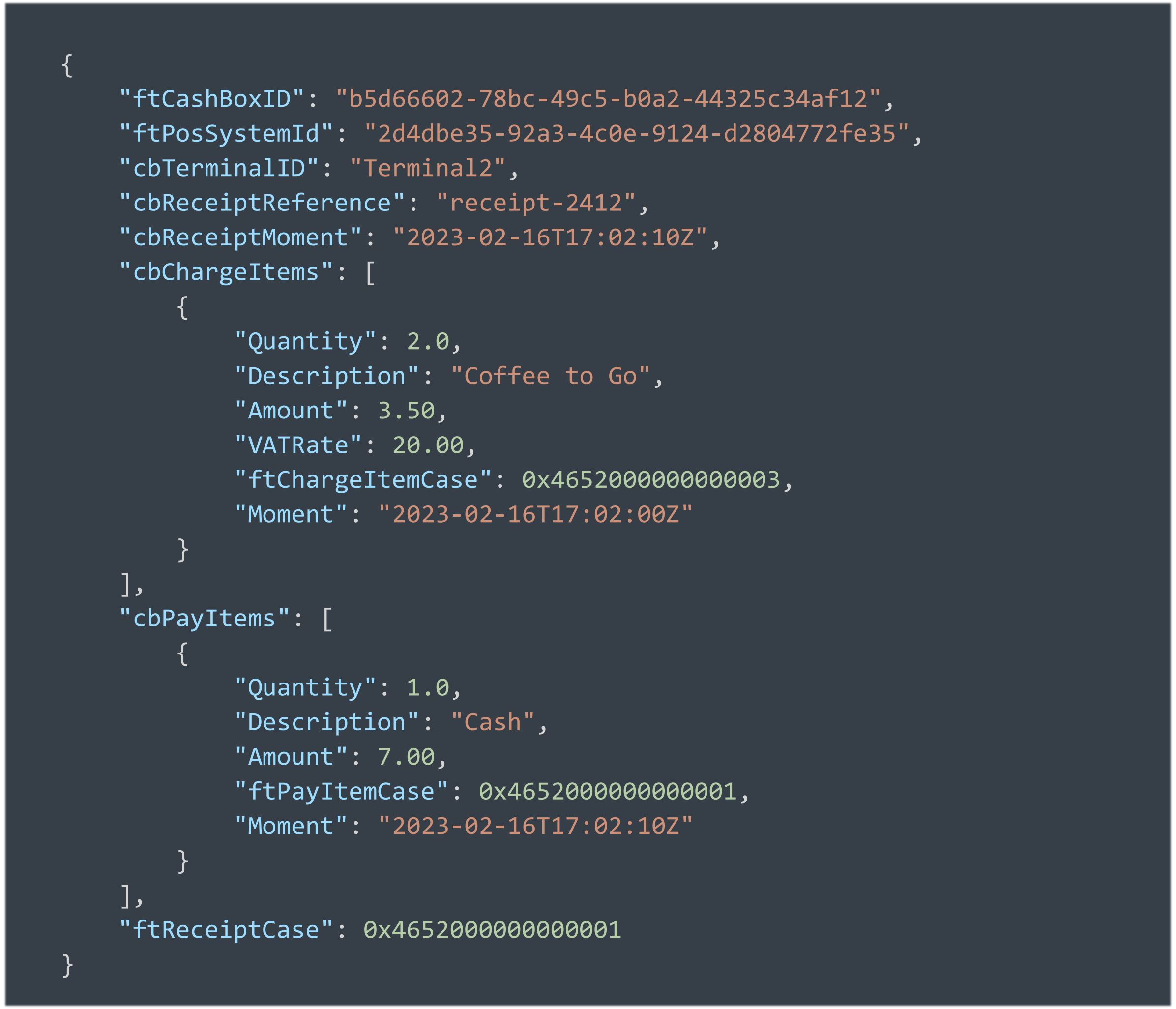
1. Execute payments with /pay
2. Fiscalize receipts with /sign
3. Create digital receipts with /print
4. Export data with /journal

… with more countries coming!

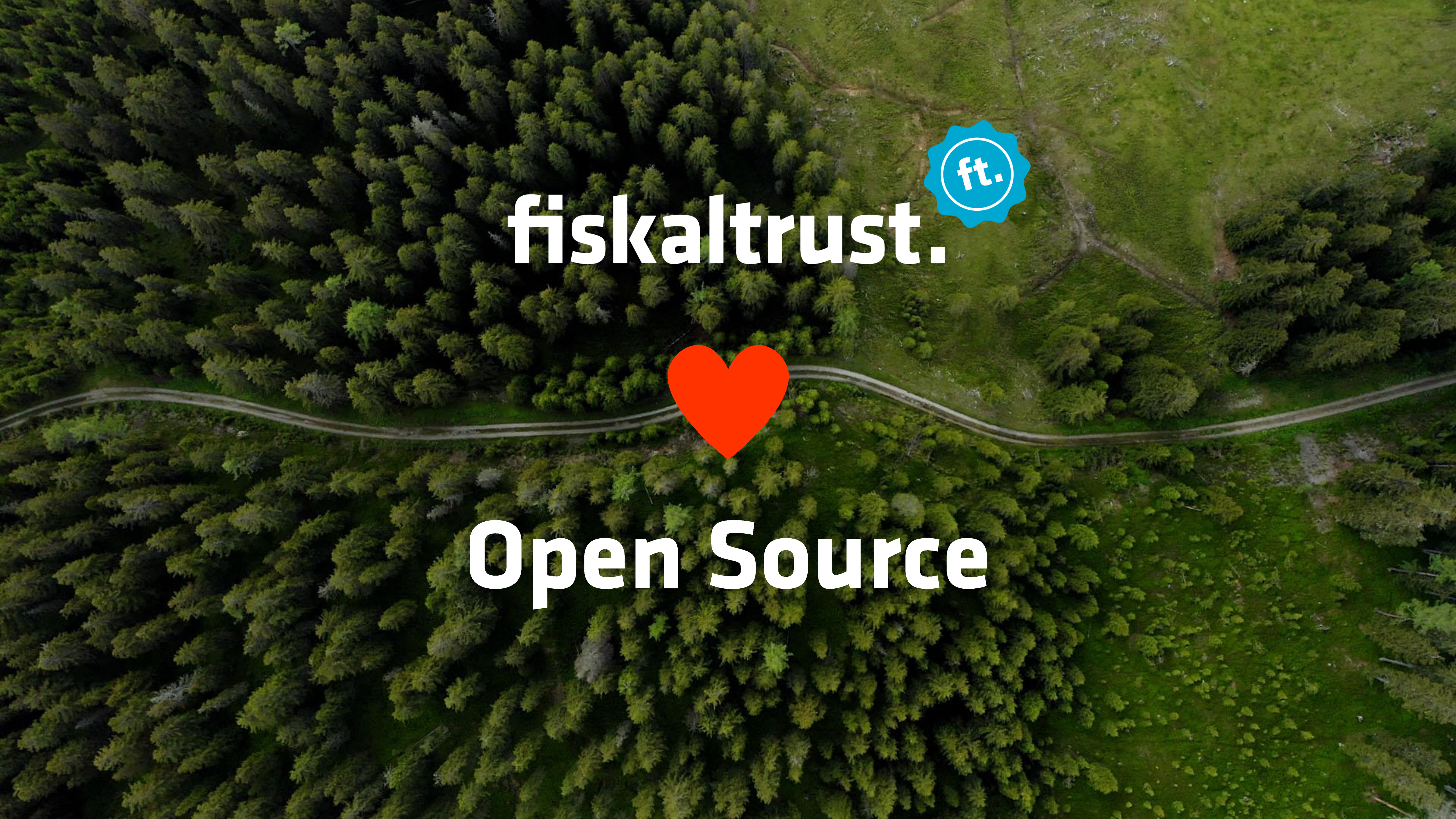
Practical example: Fiscalization in 🇩🇪 & 🇫🇷



```
{
    "ftCashBoxID": "b5d66602-78bc-49c5-b0a2-44325c34af12",
    "ftPosSystemId": "2d4dbe35-92a3-4c0e-9124-d2804772fe35",
    "cbTerminalID": "Terminal2",
    "cbReceiptReference": "receipt-2412",
    "cbReceiptMoment": "2023-02-16T17:02:10Z",
    "cbChargeItems": [
        {
            "Quantity": 2.0,
            "Description": "Coffee to Go",
            "Amount": 3.50,
            "VATRate": 19.00,
            "ftChargeItemCase": 0x4445000000000011,
            "Moment": "2023-02-16T17:02:00Z"
        }
    ],
    "cbPayItems": [
        {
            "Quantity": 1.0,
            "Description": "Cash",
            "Amount": 7.00,
            "ftPayItemCase": 0x4445000000000001,
            "Moment": "2023-02-16T17:02:10Z"
        }
    ],
    "ftReceiptCase": 0x4445000100000001
}
```



```
{
    "ftCashBoxID": "b5d66602-78bc-49c5-b0a2-44325c34af12",
    "ftPosSystemId": "2d4dbe35-92a3-4c0e-9124-d2804772fe35",
    "cbTerminalID": "Terminal2",
    "cbReceiptReference": "receipt-2412",
    "cbReceiptMoment": "2023-02-16T17:02:10Z",
    "cbChargeItems": [
        {
            "Quantity": 2.0,
            "Description": "Coffee to Go",
            "Amount": 3.50,
            "VATRate": 20.00,
            "ftChargeItemCase": 0x4652000000000003,
            "Moment": "2023-02-16T17:02:00Z"
        }
    ],
    "cbPayItems": [
        {
            "Quantity": 1.0,
            "Description": "Cash",
            "Amount": 7.00,
            "ftPayItemCase": 0x4652000000000001,
            "Moment": "2023-02-16T17:02:10Z"
        }
    ],
    "ftReceiptCase": 0x4652000000000001
}
```

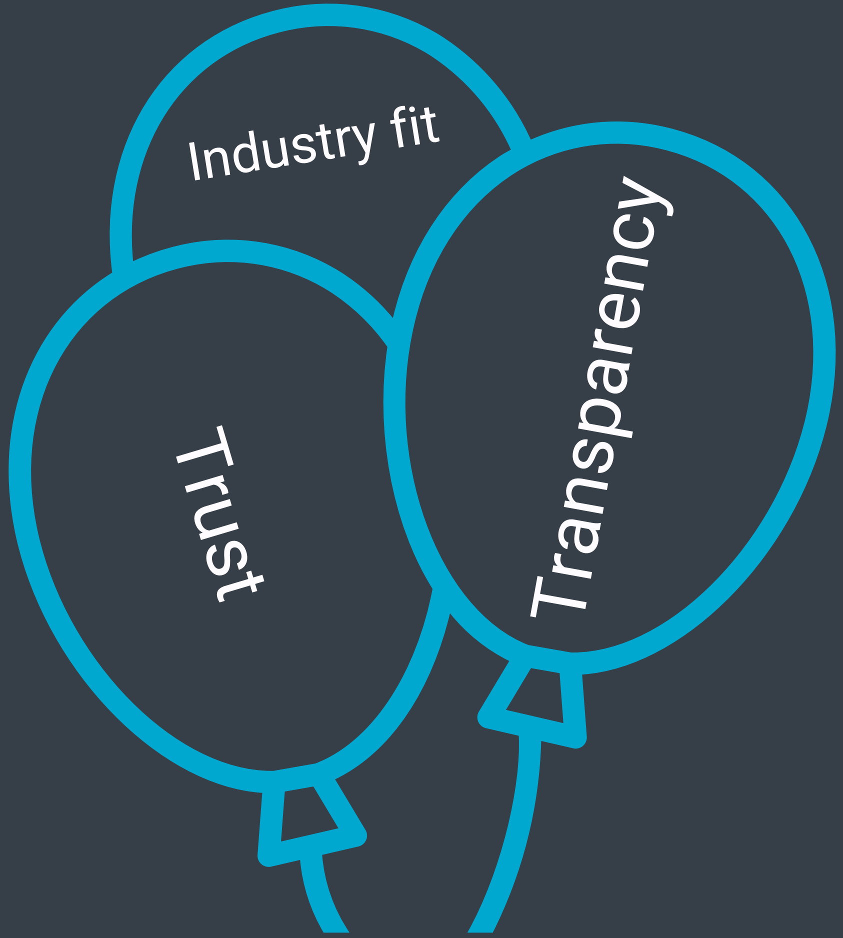


fiskaltrust.

♥

Open Source

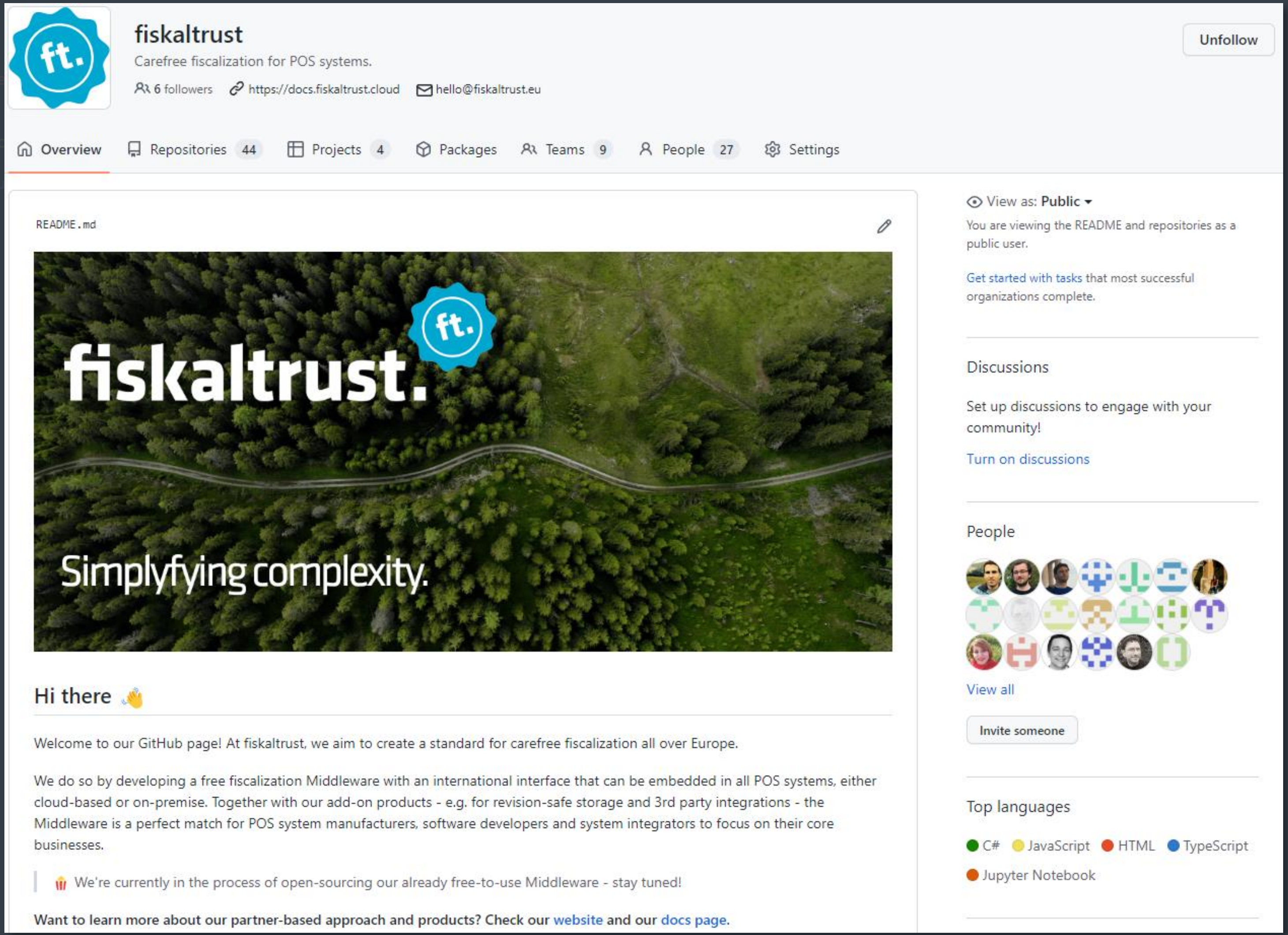
Why Open Source?



fiskaltrust.

```
c Task<ReceiptResponse> InternalSign(ftQueue queue, ReceiptRequest data)

_logger.LogTrace("SignProcessor.InternalSign called.");
if ((data.ftReceiptCase & 0x0000800000000000L) > 0)
{
    try
    {
        var foundQueueItem = await GetExistingQueueItemOrNullAsync(data).ConfigureAwait(false
        if (foundQueueItem != null)
        {
            var message = $"Queue {_queueId} found cbReceiptReference \"{foundQueueItem.cbRec
            _logger.LogWarning(message);
            await CreateActionJournalAsync(message, "", foundQueueItem.ftQueueItemId).Config

            return JsonConvert.DeserializeObject<ReceiptResponse>(foundQueueItem.response);
        }
    }
    catch (Exception x)
    {
        var message = $"Queue {_queueId} problem on receitrequest";
        _logger.LogError(x, message);
        await CreateActionJournalAsync(message, "", null).ConfigureAwait(false);
    }


    if (_receiptRequestMode == 1)
    {
        //try to sign, remove receiptrequest-flag
        data.ftReceiptCase -= 0x0000800000000000L;
    }
    else
    {
        return null;
    }
}
var queueItem = new ftQueueItem
{
    ftQueueItemId = Guid.NewGuid(),
    ftQueueId = queue.ftQueueId,
    ftQueueMoment = DateTime.UtcNow,
    ftQueueTimeout = queue.Timeout,
    cbReceiptMoment = data.cbReceiptMoment,
    cbTerminalID = data.cbTerminalID,
    cbReceiptReference = data.cbReceiptReference,
    ftQueueRow = ++queue.ftQueuedRow
};
if (queueItem.ftQueueTimeout == 0)
{
    queueItem.ftQueueTimeout = 15000;
}

queueItem.country = ReceiptRequestHelper.GetCountry(data);
queueItem.version = ReceiptRequestHelper.GetRequestVersion(data);
queueItem.request = JsonConvert.SerializeObject(data);
queueItem.requestHash = _cryptoHelper.GenerateBase64Hash(queueItem.request);
_logger.LogTrace("SignProces      InternalSign: Adding QueueItem to database.");
await _queueItemRepository.       OrUpdateAsync(queueItem).ConfigureAwait(false);
_l                    ss   InternalSign: Updating Queue in database.");
awa                          ry.InsertOrUpdateQueueAsync(queue).ConfigureAwait(false);
```

fiskaltrust.



fiskaltrust

Carefree fiscalization for POS systems.

6 followers https://docs.fiskaltrust.cloud hello@fiskaltrust.eu

README.md

Hi there 👋

Welcome to our GitHub page! At fiskaltrust, we aim to create a standard for carefree fiscalization all over Europe.

We do so by developing a free fiscalization Middleware with an international interface that can be embedded in all POS systems, either cloud-based or on-premise. Together with our add-on products - e.g. for revision-safe storage and 3rd party integrations - the Middleware is a perfect match for POS system manufacturers, software developers and system integrators to focus on their core businesses.

> 🍿 We're currently in the process of open-sourcing our already free-to-use Middleware - stay tuned!

Want to learn more about our partner-based approach and products? Check our website and our docs page.

View as: Public ▾

You are viewing the README and repositories as a public user.

Get started with tasks that most successful organizations complete.

Discussions

Set up discussions to engage with your community!

Turn on discussions

People

View all

Invite someone

Top languages

● C# ● JavaScript ● HTML ● TypeScript ● Jupyter Notebook

github.com/fiskaltrust

What’s next?

Integrate the Middleware into your POS software

Let your service be a part of the Middleware

Join the growing developer community

/fiskaltrust

/fiskaltrust

fiskaltrust.

Hall 5 / C28