29th June 2018

Assessment of the Challenges in PSD2 Implementation V2

UK API Market Release Issues and FDATA Europe Recommendations

Please contact info@fdata.org.uk for further information

Introduction

The Financial Data and Technology Association (FDATA) was founded in the United Kingdom in 2013 during the negotiations to add account aggregation to the EU Second Payment Services Directive (PSD2) and is now a trade association with a significant international footprint. The members of the association are primarily firms who deliver financial innovation to empower customers to make better decisions and take fuller control of their financial lives. As policymakers in the EU have turned their attention to the interoperability, standardisation and implementation of the various ecosystems being established under PSD2, FDATA Europe has increasingly played an advisory role with officials and stakeholders.

Through engagement with policymakers, regulators, industry stakeholders and consumer groups, FDATA has played an important role in the design, technical specifications and implementation of the United Kingdom's Open Banking ecosystem, which went live on January 13, 2018. The expertise developed over several years of being the advocate for Open Banking, through to the first significant implementation, has created the impetus for FDATA to be invited into other markets.

With its insight in Open Banking recognised globally, FDATA launched a North American chapter (FDATA NA) in 2018. FDATA NA was founded by firms whose technology-based products and services allow consumers and small businesses in Canada, the United States, and Mexico to improve their financial wellbeing. Policymakers and industry participants in North America have recently begun a dialogue around how best to balance access to third-party tools with appropriate safeguards. Through dialogue and engagement with government actors, financial institutions and third-party stakeholders, FDATA NA seeks to encourage each market to embrace Open Banking frameworks. As Open Banking spreads globally, similar FDATA chapters are currently being formed in Australia, India and other markets.

Informed by engagement with industry stakeholders and policymakers globally, and composed of leading financial and technology experts from firms of all sizes delivering immense financial benefit to millions of consumers and small businesses from all corners of the globe, FDATA is positioned to provide a well-balanced assessment of the challenges currently facing implementation of PSD2, and to offer well-reasoned solutions to improve its delivery.

What follows is a sober evaluation from a recognised expert body of the issues to be overcome in PSD2 implementation and to provide various evidence-based recommendations for improving its delivery, to ensure that the directive meets its market facing objectives of enabling competition and innovation.

1. Background

FDATA Europe held a technical workshop on the 31st May, 2018 at the Tower Hotel in London. The 30 FDATA attendees were predominantly the technical leadership of member firms, including CTOs, senior architects, lead developers, proposition owners and user experience designers, many of whom have been building services against the Open Banking API design in the UK.

Also in attendance, as observers, were some senior staff from the UK's Open Banking Implementation Entity (OBIE).

The Competition and Markets Authority order required the CMA9 to deliver 'Read' and 'Write' APIs to make Open Banking functional by the 13th January 2018. This report therefore represents a snapshot in time, one hundred and thirty six days after the initial release.

The purpose of this report is to catalogue some of the technical challenges encountered by the Third Party Providers (TPPs) during their initial connections with the Open Banking Directory and in particular thereafter with the API connections. The majority of the TPPs were focused on the AIS type connections and fewer on the PIS connections, this is as a result of both the makeup of the constituent members of FDATA Europe and also to a lesser extent reflecting the readiness of the PIS delivery amongst the CMA9 and the TPP marketplace in general.

Whilst this report is designed to focus on the challenges and issues, it should be noted that, although the TPP market that is engaging with the OBIE and CMA9 delivery have had many short term frustrations, it would be wrong to say there is a belief that the disappointment will be long lived. Some of the CMA9 outputs were well built and easy to use. The market is excited by the capability of the Directory, of consuming the API and values the work that has been put in to the dispute resolution and management capability.

Most importantly, the market values the role of the Implementation Entity and the role of the Trustee. There is a certainty with which participants can look forward to a good outcome, as the Entity has the power to solve the problems in front of it until a suitable end state is realised. Whilst other markets don't necessarily have the framework for creating such a well resourced Entity, there is no reason that they cannot benefit from its output and learnings it has thus far generated.

2. Managed Roll Out

During the process of moving the Open Banking API ecosystem into production, concerns were raised by the ASPSPs and the TPPs in respect of the adequacy of preparation and testing. In particular, the ASPSPs had significant risk issues to overcome.

The TPPs proposed and agreed to enter - on a voluntary basis - a process called 'Managed Roll Out' during which they could conduct Market Acceptance Testing, provide detailed feedback and at the same time would agree to significantly throttle the volume of PSUs that they brought into production.

Not all of the ASPSPs in the CMA9 were ready with a full suite of production APIs on the 13th of January to commence in Managed Roll Out. Despite having built to a clear specification, and having a Central Directory through which they were to expose their API endpoints, the CMA9 who were ready to deploy, had to enter into a very significant amount of hand holding with the circa dozen companies who were newly regulated TPPs. The Directory, provided by OBIE as centralised infrastructure, only permitted regulated market actors to enter the ecosystem.

During November and December 2017 the Managed Roll Out was planned. It was an attempt to substantially reduce the overall market risk and give the CMA9 some confidence in their new production APIs. These APIs would be tested with a significantly moderated volume of real customers whilst incrementally, week by week, enabling those user volumes to be increased.

The Market Acceptance Testing provided feedback both directly to the ASPSPs and also through a dedicated OBIE service desk, where tickets could be raised in relation to issues arising. This collaborative effort was essential to create the safe market conditions for this initial bedding in to take place. Without this intensively curated go live, it is arguable whether the delivery would be live several months later.

This Managed Roll Out produced a significant amount of learning for OBIE, the CMA9 and the TPP market, which is of significant value to TPPs and ASPSPs that have still to join the ecosystem, and provides materials and critical learnings for implementations in other EU markets.

3. Summary of Issues

This section will list, at a high level, the key implementation issues discovered. They will be supported with more detailed evidence later in the document.

3.1 Specification and Implementation are Not the Same Thing

Despite having a very clearly documented specification and a properly resourced implementation plan, on the 13th January the CMA9 produced:

- A self attestation of conformance to specification. This clearly failed;
- Some implementations which were closer to the specifications and implementation guidelines than others;
- Significantly different API endpoints;
- Varying level of conformance to the specified security profile;
- Materially different API structures;
- A range of different levels of conformance to the required authentication flow guidelines;
- Varying processes for helping the TPPs to connect to their APIs;
- Different error handling and error codes;
- Significantly different qualities of test environment;
- Few implementations of non-mandatory specification fields, some of which may now need to be re-categorised as compulsory.

From a learnings perspective we have to bear in mind that this was on a small and closely managed scale with only 8 ASPSPs providing API access at some stage during the Managed Roll Out. There were only a few TPPs who were permissioned from the FCA. From a TPP perspective, there were challenges in preparing for market entry in respect of production environments.

- There was too large a gap between the few test environments that were available and the production experience.

- During this period of rapid market feedback, issues were uncovered and fixed by the ASPSPs in their production environments, frequently resulting in ‘breaking changes’.

- Much of the downtime was during normal office hours and would not have been acceptable in full scale production in front of real customers.

- The JIRA tickets, which formed part of the central ticketing desk provided by the OBIE, were not ‘public’ tickets and therefore the TPPs struggled to find out how other TPPs had addressed the same issues as they entered the ecosystem.

- Some of the individual onboardings of the TPPs to the CMA9 APIs took weeks of handholding and coding workaround due to various CMA9 non-conformance issues.

- Whilst the CMA9 were very keen to receive feedback to improve their output, at times the TPPs had inadequate resources to test (often using the private accounts of staff members) and there were significant problems in the adequacy of testing through the managed rollout to enable a trend into production with real customers.

- Even at the point of this report, 136 days after commencement of Managed Roll Out, the Trustee’s Office is still forcing incremental improvements to conformance.

- There is still substantial work required to enforce strict standardisation to remove complexity and therefore to make it easier for parties to connect and to scale.

- In addition to the API and security profile conformance, the actual API performance was not consistent, with some of the APIs unstable in production and availability and rate limiting not consistently applied.

4. Key Findings

1. The CMA9 commenced the ‘read’ and ‘write’ construction work via OBIE specification in January 2017. It therefore took longer than the one year provided to complete for most of the banks, despite very significant resources being applied and the broad range of qualities produced.

2. At time of drafting the report, eighteen months after the commencement of PSD2 compliant functionality, it is estimated that perhaps 50%+ of the development requirements have still to be delivered, including some of the more challenging corporate payment components.

3. The provision of API specifications and the delivery of standardised outputs in the market are clearly not the same thing. The idea that API specification groups can create a standardised output without implementation capability is proven to be false. Attestation by any ASPSP to having built to the specification of one of the EU specification groups, is not going to work without objective technical validation. Lack of standardisation exponentially increases complexity.

4. On the one hand ASPSPs were complaining that not enough TPPs were production ready to adequately test their API endpoints whilst on the other hand TPPs were complaining of the complexity of building work arounds to connect to each ASPSPs delivery. Even at a small scale, this became very complicated very quickly. Much of the complexity was due to inconsistency in approaches betwixt and between each of the ASPSPs and also the TPPs.

5. The complexity means that the output does not scale.

6. The level of handholding generates a massive resource intensive requirement which neither the ASPSP or the TPP market can sustain. Any failure to adhere precisely to the security profile significantly increases onboarding complexity, diminishes ecosystem security due to the range of penetration tests and audit demands, make it harder for national competent authorities to validate the market competence of participants and make it increasingly difficult for the insurance market to properly understand and therefore price the threat assessment.

7. The PISP use case may have been technically easier to test but remains untested due to reliance during Managed Roll Out of testing in production. No PISPs were ready for testing due to technical or regulatory issues. The CMA9 did not adequately prepare their own test capability pre-release.

8. Forcing each bank to build a model bank - as per RTS - may improve levels of innovation capacity at these institutions, but will not be a good servant of the standardisation agenda, unless there is an intermediate stage, where the API output of the model is converged before the cross to production. It would be helpful to have experienced technical input to enhance the policy and regulatory leaders understanding of best practise in technology delivery.

9. Policy makers need to be aware that variances in the API will lead to either the TPP proposition being aligned to the minimum delivery, or that certain providers will by excluded from coverage. It is unlikely that the TPP, who is the end customer provider, will provide a service in which the customer journey is materially different depending on the ASPSPs connected, as it may lead to an incoherent user experience.

10. The UK Implementation Entity, as a non-regulated actor, actually found it extremely difficult to perform a useful role in testing and harmonisation, as it was not able to move into the production environment it had provided without regulatory permissions.

11. The substantial investment into the various conformance test tools and in the OBIE directory has made a significant difference. During the period of the Managed Roll Out, the TPP market experienced a steady reduction in the levels of technical work around required to connect up to the production environment of the CMA9.

12. Going forward, the newer TPPs will have a more standardised API delivery and find it easy to discover and connect to the end points of each delivery. There is an intention to deliver dynamic registration through the Directory to enable rapid connection and automated API mapping.

13. Quite clearly, had more TPPs been permissioned earlier by the FCA (from the 60+ applicants still awaiting permission at the time), the intensity of effort beyond the handful of actively engaged TPPs would have been completely unsustainable for the ASPSPs. For the same reasons, the Managed Roll Out would have had more testing but would have been more difficult to support and police. Again, this would not be possible without a central implementation body.

14. The RTS timetable will be difficult to reach for the EU ASPSP market. Although many of the banks of Europe are keen to be given the exemption to building the fall back, it is highly unlikely that the majority will deliver an API on time that is capable of being in a position to be provided with an exemption to the requirement to support the fall back.

15. At the time of drafting, the EBA guidelines on the RTS have appreciated that it may not be possible for each ASPSP who has built an API to a specification to also be in wide use (as the TPP market may simply not be focused on connecting to it). On the other hand, self attestation has been shown to not work.

5. Summary Recommendations

1. **Force simplicity, as the alternative complexity is unsustainable from a cost, risk, scalability and time perspective;**

2. **Forced simplicity requires forced standardisation.**

3. **Standardisation protects everybody who wants to see a good market-facing outcome for PSD2 delivery.**

4. **Don't let ASPSPs decide to build their own API specification and still qualify for the fall back exemption as it takes them out of the ecosystem.**

5. **Standardisation is not possible without both an implementation entity and regulatory environment that is capable of imposing it or an ecosystem that understands the mutual benefits of being subject to it.**

6. **The entity has a number of expensive tool sets at its disposal, which are reusable and considerably reduce the overall cost burden across the ASPSP and TPP communities.**

7. **Standardisation requires that conformance test suites need to be applied and tested on the ASPSP model bank or on some intermediate pre-production model and then also in production.**

8. **TPPs need to also be tested for conformance to the security profile as part of their regulatory journey and thereafter.**

9. **Test suites need to be applied through the point where the PSU joins, as the underlying ASPSP API quality needs to be tested to check for availability of the appropriate fields.**

10. **Optionality does not typically drive innovation in the same way that standardised outputs enable innovation, so the minimum threshold of API data payloads should be clearer.**

11. **The API performances need to be measured and published on a regular basis, showing uptime availability and any rate limiting settings.**

12. **The investment in standardisation testing tools pays off and does in the long run reduce costs, wasted time and risks, but does require an**

implementation body to deliver it and some form of independent monitoring or certification capability

13. **A directory capable of managing the local and cross border identities of permissioned actors is of key importance, as it will also enable API endpoints to be displayed in a common pattern and enable faster onboarding.**

14. **For smaller ASPSPs without the resources to be validating thousands of TPPs, this is particularly useful during the period where eIDAS is not widely delivered in all EU markets.**

15. **Having more than one directory or competing dispute management system may prove to be unhealthy competition as it will artificially create a complexity layer without adding value. This is central infrastructure to protect customers and market participants and needs to be uniform.**

16. **RTS and the EBA guidelines for NCAs need to reflect the reality of the evidence by:**

    a. **Agreeing on standards**
    b. **Converging specifications and reducing the volume of optional fields**
    c. **Creating an environment where the ASPSPs have the time to build to standards**
    d. **Creating an implementation entity - which is beyond a simple testing regime and is independent of the participants - to provide and govern the infrastructure and maintain technical capability.**
    e. **Enforcing the testing against the standards and publish the results**
    f. **Reduce the cliff edge that is the RTS date, by enabling the Credential Sharing (through to screen scraping) market to be maintained until the ASPSP has met the API test criteria, or has clearly published by the end of 2018 that they are going to build a dedicated support zone to provide access to the screen scraping with identity option.**
    g. **Ensuring that the building of standards and implementation is both lower cost and lower risk than the alternative by enforcing strict standards and making it easy for ASPSP to know that they have reached the required level.**
    h. **Improve the levels of commercial certainty for participants, by undertaking realistic impact assessments and then being explicit with requirements.**

APPENDIX 1

OBIE Authentication Flow issues

- A poorly designed Consumer Authentication Step adds considerable frustration to the PSU's experience.

- The user frustration will introduce significant drops in TPP's sign-up rates, weakening TPP's economics and cutting TPP's capacity to win customers and to deploy their model.

- The need for users to Re-authenticate every 90 days, will increase the need for these Authentication steps to be built as frictionlessly as possible.

- TPP's will be reluctant to replace screen scraping with Open Banking type APIs if the change introduces drops in customer sign-up rates.

- Current benchmarking - through deploying a/b testing models - is showing that better performing bank APIs are not materially better than screen scraping in customer conversion. Some commentators have suggested that the changing rules in PSD2 and availability off the API, and the newly regulated status of the API consuming TPPs would have encouraged more persistence in this channel.

- However, the significant conversion differences between the CMA9 can only be down to the differences in the user journeys. It is now proven that poorer authentication implementations reduce the impact of competition and the TPP community is looking forward to this being quickly resolved.

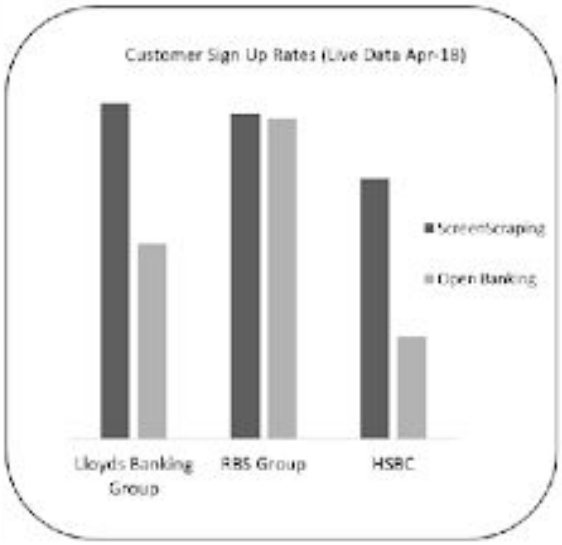
LBG, RBS and HSBC User Flow Comparison

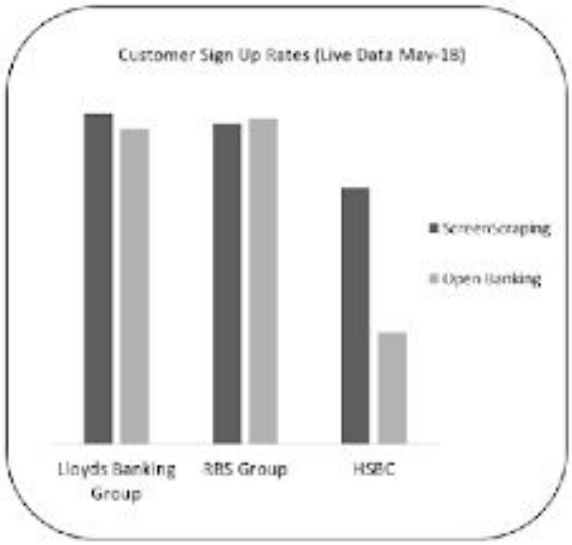
- These ASPSPs are being used because they were live in April and May.

- Negative user friction directly leads to huge swings in drop-out rates.

- The table below is demonstrating that during May 2019, RBS and Lloyds (LBG) had authentication implementations that were performing as well as the screen scraping alternative supplied by the single TPP collating this evidence (for consistency).

- We can see that whilst RBS has been consistently providing the bulk of the orders on time through April and May, LBG has significantly improved in May





- LBG has performed differently from April to May because of an extra step in the authentication flow where they had generated an automatic voice code.

- HSBC is the only one of these three banks to require multi factor authentication in the regular consumer flow. The consumer flow and business flow may not be the same – so monitoring and benchmarking capability needs to be aware of any flow variances and should keep the reporting in individual streams per flow.

    - HSBC is clearly underperforming in the conversion due to the complexity of the authentication process and for no other reason (because other factors were constant).
    - This is because HSBC is taking customers through double authentication which they don't require for data access during their own branded internet banking session with their customer.
        - It was therefore beneficial to have the comparison of open banking v. screen scraping.
    - In terms of SCA definition in RTS, HSBC could detect that the consumer is using a device that they have used before so the possession element may be considered to have been met.
    - To be clear, HSBC is not the worst performing API authentication flow, and this comparison was used because it generated enough information to plot on a graph. Some CMA9 are not being put into a live situation with real customers at point of testing due to the poor experience

- Introducing a different process for a customer going through an open banking channel (who may need a device that they don't usually carry), is creating the drop out.

- The implementation of the OBIE Consent/Authentication/Authorisation guidelines (where these have been followed) has shown that the journey does not create much friction for consumers as most just click through these steps. This is backed up by the data proof as there is no significant drop off between Screen Scraping and Open banking for the RBS Group, suggesting the extra account selection and authorisation page do not add to drop out rates.

- Action Point:
    - This analysis should extend across all the ASPSPs – regardless of authentication journey provided. Benchmarking is still useful to test the dropout points and what the barriers are. It would be useful to map it to changes in P3 (authentication evaluation at OBIE) and to develop interactive sessions to test and explain the issues.
    - Screen Scraping should be maintained as a benchmark for a/b testing for several more months.

- Screen scraping can be clunky and it is difficult to reduce friction through this process, so the ASPSP API execution should materially outperform it when it is more refined.

- Journeys which depart from the normal authentication journey that the customer experiences in their digital banking channel, is one of the main causal factors driving the delta in conversion, and the other is clunky authentication flows provided, even if they are consistent.

- Qualitatively the flow needs to be either materially and obviously better, or, if not proven as such, needs to be identical to flow the customer is familiar with.

- FDATA therefore advocates pushing for conformance to the OBIE guidelines or some set of common guidelines (that also enable clustering to improve GDPR issues) and that we seek convergence in the authentication step to a unified model, rather than leaving this in the competitive space.

- In the short term, as remedial measure, underperforming flows should be required to follow the authentication flow the ASPSPs use for their internet banking customers. For example, SCA for new payees and simplified flow for data access. The other significant benefit of introducing this requirement for ASPSPs is that it will even the competitive landscape between the ASPSP and the TPP. ASPSPs are currently incentivised to improve their online banking login flow for their direct customers but not the authentication/re-authentication flows as this benefits TPPs and other ASPSPs. The requirement outlined above would align the authentication flow with online banking login, so that improvements or deteriorations would be equally shared.***Detailed examination report of each ASPSP API authentication journey is available from FDATA***

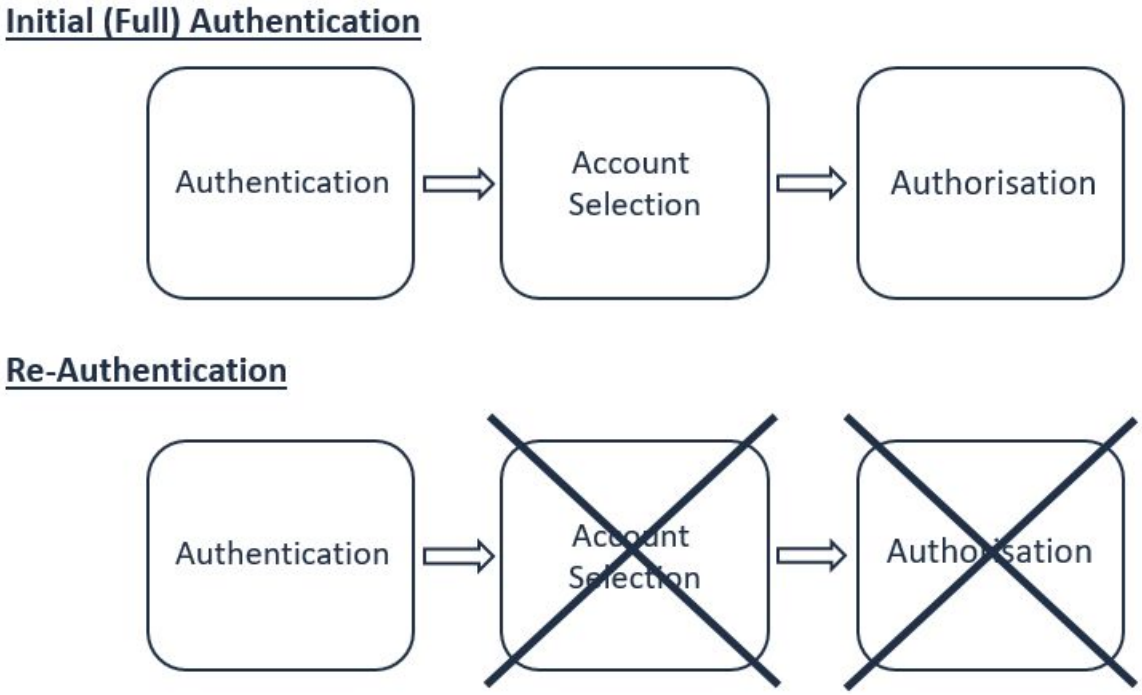
- Action Point: Need to have some focus group work to explain why consumers drop out of an authentication flow, as it is challenging to get a PSU to explain during an online session.

APPENDIX 2

Re-authentication Issues

- Main concerns of the current guidelines
    - **The re-authentication takes place at an unexpected stage of the customer journey.** A TPP is often a service provider rather than a product vendor. A PSU typically plans to use a TPP for a long period of time and is therefore unlikely to anticipate the need to re-authenticate every 90 days.
    - **Propensity for Negative User Experience -** A suboptimally designed re-authentication journey will present considerable negative friction for the PSU within the user journey.
    - Market Competition - A 90-day re-authentication cycle will negatively impact TPP's economics. This negative impact will not be shared by the ASPSPs so it remains an asymmetrical penalty.

- Re-Authentication is fundamentally different to full authentication and as such needs to be seen and treated differently.
    - For re-authentication there is no need for the user to see the account selection or authorisation screen as the account details and permissions haven't changed.



- Re-Authentication is extending the current consent by 90 days without any changes to the account selected or the permissions. Any change to consent permissions or account details will require full (initial) authentication.

- However, because re-authentication requires less steps, it can be a lot slicker. There is a difference between hard authentication and soft authentication. Hard authentication means the PSU goes through the whole process, whereas soft authentication means looking at the device and location then looking at the date they have logged on and come to the conclusion that it cannot be someone else.

- The FDATA Europe technical workshop discussed 'Passive TPPs', who provide an account monitoring AISP service which sits quietly in the background and only comes to life as an alert to signal some danger or issue arising. In which case any requirement for the TPP to initiate the re-authentication seems inappropriate and will lead to potential considerable customer detriment.

- FDATA Europe suggested re-authentication flows that could happen

- One member explored the physical device
    - Could the relationship between the TPP and the user satisfy an element of SCA? The device could be used to part authenticate.
    - If it is a new device, a 2 page process could be used to try and avoid pin-sentry interference. A text could be used to show that SCA is satisfied. This would certainly be helpful for a PSU, who is more likely to have a phone than a pin-sentry device ready to hand.

- One member suggested a consent portal on an app on someone's phone. It recognises the consent you have given to TPPs and when it needs to be re-authenticated. This could be taken further with decoupled flows.

- One member suggested that if you have used the TPP within the 90 days, this could be passed through the API as a form of validation back to the ASPSP that the service was still required.
    - There has to be communication to the PSU – if they have not logged on - from the TPP that they are about to be switched off, they need to have sufficient time to encourage them to response.
    - This could be taken further with decoupled flows.

- One member suggested that rather than individualised SCA for each TPP, that during the login to the ASPSP that is closest to 90 days, that the ASPSP provides an 'in force consents dashboard', which forms deemed consent for the next 90 days unless the PSU switches off access for a specific TPP. To reduce friction if the PSU has many TPP connections, it would be helpful.

- There is a risk, however, for the TPP in handing over all responsibility to an ASPSP, so there needs to be a mechanism for re-authentication to be started by the ASPSP and the TPP.

- Regulators are now keen to see a work around to this 90 day issue.

- If the 90 days was to be increased, this would probably need to gain an adjustment to the RTS and therefore may require a policy approval by the European Parliament, so in the short term some technical workarounds will be require

The requirement for 90 day re-authentication is fundamentally materially damaging to business cases and economic viability of AIS models. If the market wants this sector to flourish, it needs to have its impact materially reduced and then completely removed. The PSU should dictate the term of Consent, and the TPP, as a regulated actor, should have a requirement for communication to be made to remind the customer of their connection. It would be helpful if the EBA/EC could conduct a detailed impact assessment, collecting evidence of the likely TPP business failure rate directly attributable to this requirement.

APPENDIX 3

Data Artefacts

ASPSPs in the CMA9 have generally delivered what is in the mandatory specification but have primarily failed to deliver optional fields. There's more to do to improve API payloads.

This section examines key areas identified as needing to be improved to ensure Open Banking is capable of solving the market facing problems the TPPs wish to solve. In particular, the data payload are failing to deliver

- Unique Transaction Identification (UTI)
- Transaction Metadata
- Personal Data

Transaction description is an optional field. This should be made mandatory. OBIE has specified this from March 2017,, but did not make it mandatory at the time because it is not a regulatory requirement.. FDATA will table a Change Request, seeking to have this redesignated as a requirement rather than an option for Version 3

Unique Transaction Identification (UTI)

- TPPs collect transaction information periodically via the AISP APIs. In many use cases TPPs will "merge" transactions with existing transaction sets.

- Without UTIs, one must attempt to work around specific problems when "merging" changes to prevent **duplicate** or **missing** transactions being processed.

- Contrived illustration of this problem during pagination follows.

- Most ASPSPs don't provide a UTI to their PSU, so it is not considered directly in PSD2.

- Some banks complain that they don't actually have unique transaction IDs, despite the obvious benefit in evidence tracking and fraud prevention. A key issue is that sometimes booked and pending transactions exist on different systems within the bank.

- Experienced screen scraper developers can overcome this issue with transaction syntax and a large volume of data (to provide edge use cases to train with), but this is not 100% accurate.

- However, transactions can change historically and there can be many duplicates. Pending transactions are a lot more likely to change if booked ones have unique IDs. If the banks are in the process of putting IDs in the data-lay then this is probably something that they do not need to apply fuzzy logic to. It is just splitting from pending

logic to data lay. Aggregation services have difficulty in removing fuzzy logic. Having an ID would help reconcile between ‘posted’ and ‘pending’ transactions.

- Occasionally, booked transactions can change.

- It doesn't really matter what the format of the UTI is, as long as it unique.

- Transactions ledgers are not an immutable ledger of history. Transactions get inserted and removed and others get posted without a time and date stamp. Nationwide show the post data the day before the money moves. Halifax gets posted a day after the money moves.

Why do we need UTI?

- A TPP must process accurate transaction data in order to provide reliable services for PSUs that don’t miss or duplicate transactional data. Incorrect data will drive alerts that should not be triggered or cause alerts to fail when they should be triggered, or falsely identify available money when it is really spent, or lead to financial product decisions from distributors to be made with false evidence.

- Additional benefits
    - TPPs can reference resources via standardised immutable identifiers in order to disambiguate resources when troubleshooting issues with ASPSPs
    - TPP can have assurance that they have already processed a subset of resources so they can avoid paginating through the entire resource result set.
    - TPPs are keen to encourage ASPSPs and OBIE to develop additional standards and endpoints that might interact with resources on an individual level.

- In reality, there are no ASPSPs implementing transaction ID or other resource identifiers.

Since UTIs are currently optional in the specification and no banks are implementing them, there are other potential solutions that were discussed:

- Imperfect solution – PostedDateTime, Amount, Description
    - But the BookingDateTime must be guaranteed to be precise which it is not (so this imperfect solution would not work)

- TPPs could use the ASPSP updates for this purpose. When a bank updates the ledger you can use fuzzy matching but you may have to ‘double hit’ the API. The time the ASPSP inserts it into the ledger is irrelevant if it is updated.

- Even in the screen scraping world they go back every 14 days and get the transactions from the last 90 days just to lock down the final position.

Technical Drivers Of UTI

The lack of requirement for UTIs directly results in TPPs requesting more transactions than would be necessary with Transaction IDs. Without UTI there is no way for a TPP to accurately de-duplicate transactions, therefore for TPPs to match exactly what a user sees through their ASPSP's portal they would request all of the user's transactions. This is not ideal for several reasons:

- Comes at significant cost for the TPP and ASPSP.
- Will increase the amount of time a GetAccountTransaction Request (Refresh) takes to complete.

If a TPP decides to limit the size of their GetAccountTransaction requests by implementing matching logic, every time this logic did not match the banks it could cause customer dissatisfaction. As this risk of customer dissatisfaction is not shared by the ASPSPs, an argument could be made that this is not promoting a fair and equal framework for competition (the key principle of Open Banking). Several example scenarios where transaction IDs would detriment users follow.

By making transaction IDs a mandatory field a TPP would request and store only the new transactions that do not match any transaction ids the TPP has stored previously. This would reduce the size of the GetAccountTransaction requests TPPs are making and would remove the possibility of negative customer experience when a TPP's transaction matching logic does not work correctly.

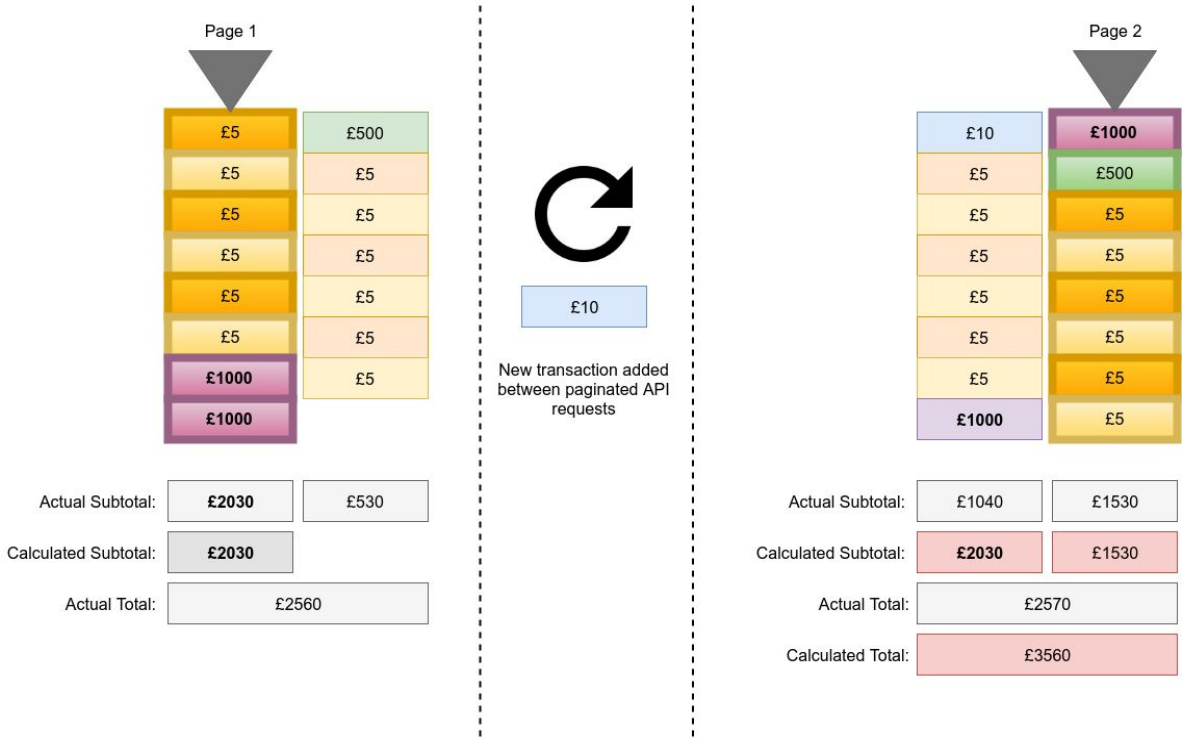
When Open Banking was launched, several ASPSPs expressed the need for TPPs to follow the principle of Data Minimisation. For for this to be achieved, transactions IDs need to be made mandatory.

On discussion FDATA recognise the difficulty for ASPSPs to create immutable and unique transaction IDs across pending and posted transactions and express the need for mandatory transaction IDs where the transaction is posted.

Example of User Detriment without Transaction IDs

High Usage Accounts

There is the possibility that a transaction is posted during the GetAccountTransaction request. Although this is unlikely for personal accounts, for high usage accounts (such as business accounts) this is very likely. In this case a new transaction is added to the ASPSP's data which does not come through during the GetAccountTransaction request, this would case all the other transactions to shift position. This would mean a transaction could shift from the ASPSP's page between paginated API requests. This would cause the transaction to be duplicated as given in the image below. Without transaction IDs there would be no way for the TPP to notice this as a duplication and would consider these transactions as two separate cases.



If the TPP is monitoring the account for product affordability purposes a duplication of a salary transaction or a high profile bill would easily invalidate the affordability decision.

Changes to Transaction Descriptions

Without Transaction IDs any change to a transactions description could break the TPP's matching logic causing a duplication of the transaction. Examples of cases where a transaction's description has mutated are given below.

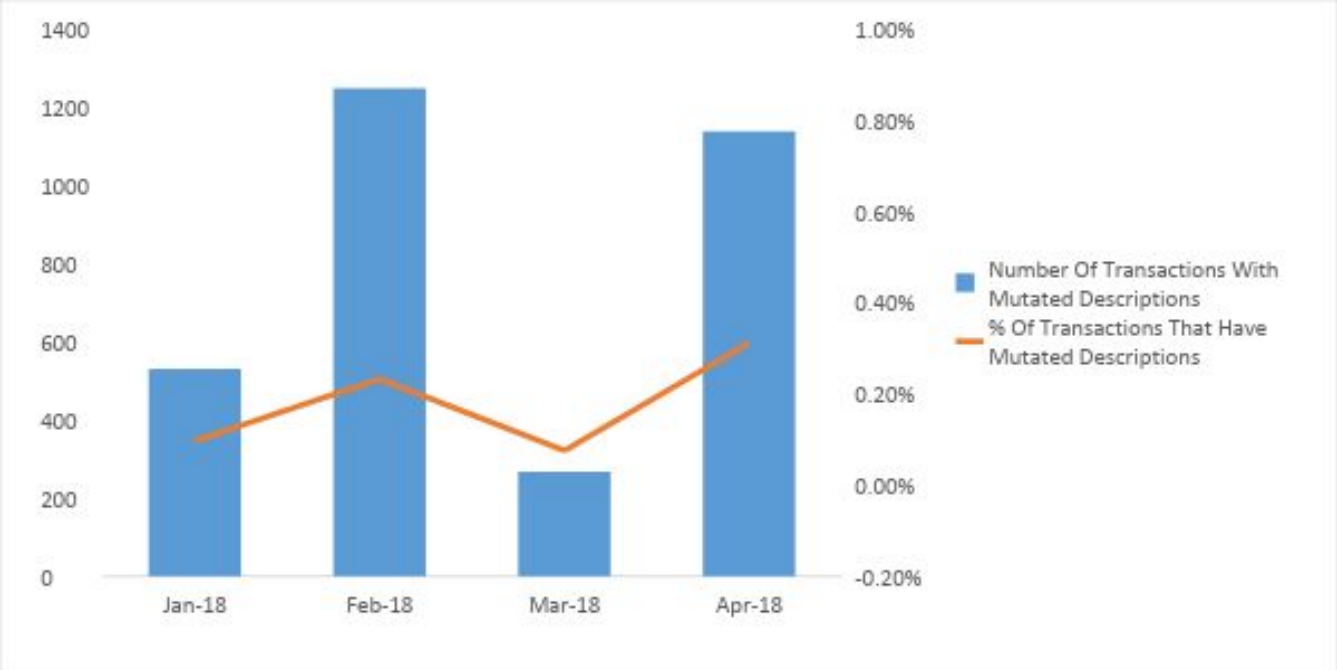
Transaction Request 1:

TransactionDescription - 1	Amount	TransactionType	TransactionStatus	postdate
XXXXXXXX & XXXXXXXXXX NATWEST	50	credit	posted	2018-01-01
Contactless Card Purchase PRIMARK STORES	1.8	debit	posted	2018-01-01
Credit E.ON XXXXXXXXXXXXXXX	76.5	debit	posted	2018-01-02
Counter Credit XXXXXXXXXXX TRANSFER XXX	1	credit	posted	2018-01-02
Card Purchase PAYPAL *PROFITABLE XXXXXXX	34	debit	posted	2018-01-02

Transaction Request 2:

TransactionDescription - 1	Amount	TransactionType	TransactionStatus	postdate
XXXXXXXX & XXXXXXXXXX NATWEST - CR	50	credit	posted	2018-01-01
PRIMARK STORES LTD Contactless Card Purchase	1.8	debit	posted	2018-01-01
E.ON Direct Debit XXXXXXXXXXXXXX	76.5	debit	posted	2018-01-02
XXXXXXXXXXX Counter Credit TRANSFER XXX	1	credit	posted	2018-01-02
PAYPAL *PROFITABLE Card Purchase XXXXXXXXXX	34	debit	posted	2018-01-02

One of our members has put together a view of the number of transactions where the transaction description was noticed to mutate. It is worth noting that this list will be smaller than the total list as there will be cases where the mutation was not picked up on.



Transaction Metadata

- Transaction Metadata includes merchant data and category codes.
    - Screen scraping inspects elements of a page which have metadata filled in.
    - The more contextual the information there is about a transaction the better
    - All ASPSPS have metadata but are not necessarily reporting it. *Aggregation Services have provided FDATA with their list of available fields.*

- All TPPs enumerate the available transactions, but many fields are optional and other fields non-standardised.

- As this Metadata is provided by the ASPSP to their users there is a regulatory requirement that it is provided to the TPPs.

- There is a real feeling that ASPSPs have provided “the minimum” when it comes to AISP v1 implementation - understandable given time constraints and existing delivery date slippage
    - This now needs to be improved in V3.
    - For transactions this could be conformant with OBIE V1 specification. This does not feel as though it is in line with regulation. If it is booked and it would appear on a statement then that is more workable.

- Merchant information
    - Invaluable for Categorisation systems, PFM etc
    - Gives TPPs the power to easily breakdown consumer spending
    - Augments “Transaction Information” field to enrich the narrative around each transaction which in turn empowers TPPs to create more personalized offerings using the additional context.
    - This is not exposed by the API but there is some more expansive description in the transaction
    - This has not been used in any current CMA9 ASPSP implementation yet.

- Category codes
    - Standardised and proprietary transactions.
    - Give richer context about the type of transaction, not the who. But otherwise, as above.
    - Transactional context is everything. TPPs compete on their ability to provide better context/categorisation.
    - Any ASPSP has this information which they could choose to expose
    - **Action Point – could this potentially expose major beneficiaries?**
    - ASPSPs are expected to provide proprietary bank transaction codes – this is found in your bank statement like debit or transaction.
    - Standardised transactions tend to go into more detail.
    - These are not enforced at a technical level, and it is not part of the specification. The v1.1 API makes it clear that ASPSPs were expected to do it, but it was never enforced as a technical decision around the validation tooling.

- If this is optional, banks call it different things. Making it standardised would bring it all into one formalised definition..

- In addition to ‘Mandatory’ and ‘Optional’, FDATA recommends the creation of a new category ‘Mandatory if Available’ to make clear that the ASPSP has to provide it if it is available for them to provide.

- ‘Optional’ should therefore be reserved for things that the ASPSP could provide to improve functionality if they wish to.

Solutions

- Policy level requirement to implement ‘Transaction Code’ for conformance (as hinted at in v1.1 spec)

- Encourage ASPSPs to implement Merchant Information where available.
    - Not sure how feasible this would be to mandate - some ASPSPs may not *have* this information available. This may need regulatory pressure.

- There are two options for the ASPSPs
    - Mandatory – the information must be passed over to the TPPs
    - Mandatory if available – If you store this data you must provide it. If you do not collect it, you cannot provide it. However, many ASPSPs think that optional means that even if you do have it, you can still choose whether to pass it over. This language needs to be clarified
    - OBIE have stated that if you provide information to a customer it is a regulatory requirement to be given to a TPP as well. These are often provided in an ASPSPs consumer interface - as such should they be reflected in a programmatic API if it accurately meets the “Account Information” definition?

Personally Identifiable Information (PII)

Why do we need it?

- The AISP API provides a rich representation of a PSU’s accounts that they have consented to.
    - If there is someone with an account request – matching the identifiers of the PSU can be very valuable in reducing fraud and introducing a smoother onboarding process in financial distribution. You can tether PII to the sort code and account number.

- However in a number of use-cases where data is aggregated, it is important to ensure that this representation is the right PSU.
    - Not getting a name and address through API could pose a fraud risk.
    - Also, many banks will not provide PII for APIs but will for beneficiaries.

- As such it's important to the TPP community that account information is provided with the identity such that TPPs can be assured of which PSU they are dealing with matches the account they are getting from the ASPSP.
    - Improvement for KYC checks.
    - Without Personal information TPPs are exposed to fraud.
    - Applications information on the TPP side and transaction data from the ASPSP – it can be difficult to match these two entities, they can make assumptions but this cannot be made certain.
    - If all ASPSPs had this on their digital delivery to their customer then it would reduce fraud in the customer base and fraud against the PSU. Much of the PII is currently available via Screen Scraping
    - It is important for:
        - Credit Reference Agency - in order to provide our fourth party (lenders) the data they need to give the PSU access to the right financial products, we need strong assurance that the PSU **is** our customer.
        - Identity and eKYC
        - AML and counter-fraud opportunity is huge.
        - Any product that combines financial data with other data sources is going to benefit.
        - If identity is not provided 3rd party solutions will be needed that tie ASPSP provided data with the user - this will create a poor user experience.

- V2 –party endpoint
    - The "Party" endpoint as specified in V2 OBIE specification provides much of what is required
    - Implementation remains optional and it seems likely that the trend of optional endpoints not being implemented will continue, unless the new mandatory if available definition is introduced.

- Challenges and Roadblocks
    - It is obvious that ASPSPs express concern over becoming an "identity provider" that performs expensive identity verification services on behalf of an extensive community of TPPs that "leech" it for free. However, all benefit from a reduction in fraud. Many ASPSPs will be TPPs and will learn the importance of these artefacts.

- Legislative alignment
    - This information is provided in many ASPSPs customer interfaces - replacing this with API workflows is necessary to deprecate screen-scraping entirely.

The ASPSP will always have structural competitive advantage if PII is available to them and not to the TPP when the ASPSP is operating also in the TPP role. Given that many of the current TPP business models are entirely reliant on PII via screen scraping without identity, under the current guidelines which classify this data as not being account information for the purposes of PSD2, the ***TPP will either fail or be required to use GDPR portability and continue to screen scrape (without identity).*** *This is allowed because it is out of scope for regulation. It is also nonsensical.*

Given that this PII is used to federate identity to improve speed of onboarding and customer conversion, whilst also delivering fraud reduction, it would be sensible for the entire market to review whether this should be reclassified and made mandatory.

APPENDIX 4

CMA9 API Quality

Compared to the performance of the API ecosystem in January 2018, there has been real improvement in the delivery. There are now minimal variations.

Rate Limits

- The rates should be fair, proportionate and mostly transparent. If the bank applies rate limits, these should be communicated to the TPPs. There needs to be transparency in rate limits.

- RBS has a rate limit. Not enough data is available to provide much detail on the limit, only that it limits refreshes to slower than screen scraping. LBG published a rate limit erroneously, but did not actually have one.

- **Action point**: Where possible look at the boundary of the rates, just because the API message came back with an error code it does not necessarily mean that this is the impetus of the rate limit.
    - This action point was queried as TPPs should tend to believe what the bank is telling them, if there is an error code then surely this should be seen as such.

- The general consensus was that TPPs are happy for rate limits to be implemented and actually feel like there could be a need for ASPSPs to implement them. However these rate limits need to be built to prevent system failures, rather than to limit TPPs functionality. On top of this there may need to be clarity on what the ASPSP is setting the limit to and a minimum accepted limit.

Error handling

- TPPs have really struggled to understand the error codes generated when the API fails in some way.

- Generic error codes lead to confusion.

- TPPs wish to see consistent, accurate and descriptive error codes to ensure that the problem resolution is faster and the monitoring function empowered with real information.

- **Action point**
    - OBIE to make error handling consistency a priority and increase the monitoring function.
    - TPPs to report any inconsistency

CMA9 provided dedicated teams during the Managed Roll Out to support error handling and invalid request reports. The TPP findings as follows:

- The CMA9 support teams came back very quickly but needed significant briefing to understand how the issues presented.
    - Needed to have multiple conversations to resolve these issues.
    - Needed better diagnostics with better escalation processes.

- Feedback to the OBIE is that it is not quite a production environment. Is this to be defined by the ASPSPs?

- Need to focus on better ecosystem cooperation coordinated by the implementation entity. Better sharing of issues and support tickets would generally enable the knowledge base to improve and make it easier to solve issues with permanent fixes.
    - Need to develop a method of sharing similar issues, unless related to specific coding problems.
    - Time issues would be helpful to share – error codes are not granular enough to be properly implemented.

Conformance Test Suite

- OBIE has produced a Conformance Test Suite to test the security profile conformity and availability of standardised API endpoints of the CMA9 banks.

- This is of vital importance to implementation. Unfortunately many of the banks had not run it in January. It appears that the non-conformance was only discovered in full production environments. The entire ecosystem needs to support and embrace the conformance programme to reduce complexity, risk, build time and maintenance.

- The level of resource required to enable connections with CMA9 banks who had not run and proved conformance was extreme, and in many cases took weeks of support.

- TPPs need the ASPSPs to meet the specification in the pre-production environment and in production.

- To help the regulators, ASPSPs and the overall security and liability requirements, it would be equally sensible for the TPPs to also have a TPP Conformance Suite developed.

- What is a conformance test?
    - OBIE has built and is building further software that ASPSPs can use to check all the security parameters and API calls. The first iteration of this enables a test of the security profile conformance using an automated and therefore easily

repeatable test framework. The test framework creates a score sheet for the firm running it.

- The further ASPSP tests will also go beyond the security profile and test the API fields and API conformance and perhaps test the availability. OBIE should extend it to testing the pre-production environment.
- A conformance test suite for the TPPs would be very welcome. This industry should discuss the value in making the TPP test suites mandatory. Some TPPs would be concerned about additional work. On the other hand, it is easy to make an argument that if the ASPSPs go through it and the TPPs therefore get a standardised outcome, that the very significant reduction in costs and risks would make an incentive to participate in this cooperative ecosystem more appealing.
- Test Suites should be run very regularly. The scores for ASPSPs and TPPs should be available to the Implementation Entity and also (going forward) to National Competent Authorities.
    - In the future, this could become part of the required reporting and improve security architecture of the ecosystem
    - OBIE should make information on ASPSP non-conformance available to the TPPs, if updates change the profile of the API connection.

APPENDIX 5

OBIE Monitoring Capability

The OBIE Trustee is setting up a monitoring function.

It is important for TPPs to focus on helping to frame the reporting requirements relating to items that need to be quantitatively and qualitatively measured in support of the monitoring function. OBIE is aiming for a world class solution and best practise of sharing information to enable rapid improvement will be applied where it does not interfere with security.

- Security Issues arising.
- Availability and any downtime.
- API response speeds.
- Any rate limiting.
- Available fields in an API
    - Conformance results
    - Availability of mandatory and non-mandatory data items
- Aggregation of key JIRA tickets and regular publication.
- Tracking and publishing all ASPSPs and TPPs in the ecosystem.
- All tickets that are raised with ASPSPs by TPPs should be raised with the OBIE service desk.
- Error Code consistency and conformity.
- TPPs to actively make suggestions of other key points to track. FDATA Europe will coordinate the capture of monitoring requirements.

APPENDIX 6

Testing and Test Planning

PSD2 RTS

- Article 30 (5) defines the requirement for ASPSP's to provide a "**testing facility**"
    - "Account servicing payment service providers shall make available a testing facility, including support, for connection and functional testing to enable authorised payment initiation service providers, payment service providers issuing card-based payment instruments and account information service providers, or payment service providers that have applied for the relevant authorisation, to test their software and applications used for offering a payment service to users. This testing facility should be made available no later than six months before the application date referred to in Article 38(2) or before the target date for the market launch of the access interface when the launch takes place after the date referred to in Article 38(2).
    - However, no sensitive information shall be shared through the testing facility."

- This needs to be implemented on the 14th March.

- One interpretation is that you get 3 months' notice to test.

- RTS is vague on testing periods and how notice should be given. When do the test environments get the version of the code that is viewed as semi-static and expected to go live in three months.
    - When a new version is published they have to provide previous versions.
    - Before any change, they can look at the sandbox before it is put into production, which will provide a new interface to work against.
    - The RTS does not specify how to give notice – only when – but the interpretation is that this would be handled and that banks would introduce a new version, but before they had turned the previous version off, they have to give three months' notice.
    - In an emergency situation, an ASPSP will document the change and make them available to the competent authorities.
        - What sort of emergency would need for you to break it? A new vulnerability – sometimes when something is completely fine then it breaks it may need to be fixed overnight.
        - Release management can prove difficult as there is not one company writing its own software for its own customers, this is about ASPSPs introducing new functionality that lots of TPPs and therefore lots of PSUs are connected to.
        - Many ASPSPs have never had that 3rd party software relationship where they have rapidly changing circumstances.

    - No sensitive data should be shared through testing processes, so it is challenging to pre-test production. If the ASPSP works it to production and its testing is being completed in production in this time window, TPPs cannot build against it. This is unworkable for TPPs,who need a solid-state period to test and build against, without testing of the ASPSP being the inhibitor.

- This is clearly a critical challenge for RTS objectives. There really needs to be a pre-production environment that is capable of undertaking the conformance test suite, to ensure that in production environment is driving down time into the TPP build whilst the ASPSP finishes testing.

UK OBIE Directory Sandbox

On first release this supported Multiparty Industry Testing (MIT) for TPPs who (by definition) were not AISP or PISP because PSD2 had not come into force.

- Some ASPSPs had close to production data, others provided completely made up data sets and the functionality needed to be improved.

- OBIE introduced the concept of model banks – these exist in the model sandbox. This enables TPPs and ASPSPs to be in whatever testing environment they want and allows access into the RTS sandboxes that the ASPSPs interact with.
    - In this case, it does not matter if a developer uses agile or waterfall as a methodology, they can still access the sandbox.

- OBIE has asked FDATA for help in defining what a testing environment looks like when a specification and a technical transition plan to go live (in a standards based environment) is done really well.

- Discovery use case –
    - Experimenting with the authorisation steps, without going near sensitive data.
    - Ensure that when genuine interactions are created, that it can be used within this environment.
    - Testing ASPSPs APIs. To do this effectively you need:
        - To ensure it is always on
        - To ensure that it is representative of production systems so that authentication flows are the same in production systems and the data fields within those accounts is also representative of production..
        - TPPs to help prepare the use cases for the testing facility
        - Proper documentation or guidelines to be provided.
    - New environments will need to be as backwards compatible as possible

- **Timing is a deep and serious problem**
    - Not every bank will introduce new delivery to the testing environment to meet the Standards specification simultaneously. There needs to be a sufficient support framework provided by the implementation entity to manage communications, timing and the rhythm of the development and release cycle. For the TPP to adopt and deploy in the market any new functionality will require substantial coverage to be available at the same time. It is important that any market entry ASPSP joining, comes into the release time of the current standard, and not building to an earlier version.
    - Much work needs to be put in to make the ecosystem easy for new ASPSPs and TPPs to join at a current stage. The ability to scale the volume of market participants in the API ecosystem is critical, if the downside risks of fragmentation and complexity are to be avoided.

Model Bank or Sandbox Guidelines

- Many ASPSPs want to know what the sandbox requirement of RTS actually means as a practical step.
    - FDATA is not certain that it is helpful unless an intermediate step is used to conformance test, otherwise we could artificially drive fragmentation in production
    - It could just be that it is not just the code that is changing (including the regression testing) but the process the ASPSPs are building, mutates to the extent that it does not provide them with the same live instances.

- Is the requirement to have each ASPSP produce a Model Bank useful in supporting the market facing outcomes of PSD2?
    - ASPSPs often have model banks to enable innovation.
    - TPPs do not want to stifle innovation.
    - Should there not be one PSD2 sandbox per release, with associated conformance test suite, so that when each ASPSP runs testing it will improve alignment rather than promote fragmentation? If every ASPSP is required to build a sandbox based on guidelines and specifications and then the production is supposed to be identical then it is obviously more difficult and more complicated.

- FDATA suggests doing both
    - There is a need to have a modern central pre-production sandbox
    - Each bank could choose just to use the central sandbox to reduce cost and complexity
    - Other banks will prefer to also have a sandbox to test out different improvements and show the options to the standards group or directly to the market as additional features. They would still go through the central pre-production sandbox before going into production.

- RTS does not specify harmonisation of the sandbox. The evidence of the OBIE implementation shows that some harmonisation at this level is a requirement to make a lower risk and standardised production environment with sufficient time for both testing and building against.
- It would be helpful for this the be at least an available and certified EU level sandbox, with the capability of extending to the other jurisdictions as required
- Given the time frame, requiring all ASPSPs to build a unique version is both unhelpful and also extremely unlikely.
- There cannot be a one size fits all form of testing due to legacy systems in the ASPSP community. But if a centralized sandbox is pushed for then it will put banks and TPPs in the same position.
- If the ASPSPs go through their own sandbox, they can load it with model banks. The model banks in production today are diverse and need more variability removed, so there needs to be a mix of the sandbox and the conformance suite. The authorities need to understand that without this, the delivery is not scalable and would have TPPs having to engage in thousands of sandboxes at different stages of evolution.
- **The Central Sandbox could be also formed for each version of the standard on a cloud instance, and then replicated for each ASPSP, so they don't have to build their own**
- It needs to be easy for a regulator to see what a good sandbox looks like, and ensure that this is the standard that is being built.
- The security profile must be standardized
- The functional elements within the security profile need to be programmatically discoverable.
- **Action Point** – there needs to be a definition of what a good sandbox looks like and help is needed to get there.