



USEF Flexibility
Trading Protocol
Specifications

INTELLECTUAL PROPERTY DISCLAIMER

This specification is provided to you “as is” with no warranties whatsoever, including any warranty of merchantability, non-infringement, or fitness for any particular purpose. The authors of this specification disclaim all liability, including liability for infringement of any proprietary rights, relating to use or implementation of information in this specification. The provision of this specification to you does not provide you with any license, express or implied, by estoppel or otherwise, to any intellectual property rights.

USEF Flex Trading Protocol

The Universal Smart Energy Framework (USEF) has been developed to drive a fast, fair and lowest cost route to an integrated smart energy future. By delivering one common standard, it enables cost-efficient connectivity of all smart energy projects and technologies. Its market structure, roles, rules and tools for the commoditization and trading of flexible energy usage work with existing energy markets.

The USEF Flex Trading protocol (UFTP) is a subset of the USEF framework. Focused specifically on the exchange of flexibility between Aggregators (AGRs) and DSOs, it describes the corresponding market interactions between them to resolve grid constraints by applying congestion management or grid-capacity management. UFTP can be used as a stand-alone protocol for flexibility forecasting, offering, ordering, and settlement processes.

UFTP and the USEF Framework were originally developed, maintained and audited by the USEF Foundation. In 2021, USEF moved to the Linux Foundation Energy, and the project is rebranded as ‘Shapeshifter – a universal flex trading protocol’.

Table of Contents

| | |
|---|-----------|
| USEF Flex Trading Protocol | 3 |
| Table of Contents | 4 |
| 1 About UFTP | 5 |
| 1.1 Purpose | 5 |
| 1.2 Scope | 5 |
| 1.3 Levels of compliancy | 6 |
| 1.4 Reading guideline | 7 |
| 2 General description | 8 |
| 2.1 USEF general concepts | 8 |
| 2.2 Contract phase | 10 |
| 2.3 Plan phase | 13 |
| 2.4 Validate phase | 14 |
| 2.5 Operate phase | 21 |
| 2.6 Settle phase | 22 |
| 3 Use case descriptions | 26 |
| 3.1 Scope | 26 |
| 3.2 Privacy and security by design | 26 |
| 3.3 Common concepts | 27 |
| 3.4 Reading guidelines | 27 |
| 3.5 Message validation | 27 |
| 3.6 Use Cases – Contract phase | 28 |
| 3.7 Use Cases – Plan phase | 31 |
| 3.8 Use cases – Validate phase | 34 |
| 3.9 Use cases - Operate phase | 40 |
| 3.10 Use cases - Settle phase | 41 |
| 4 Message descriptions | 44 |
| 4.1 Scope | 44 |
| 4.2 Message catalog | 44 |
| Appendix 1 Glossary | 65 |
| Appendix 2 Rationale for information exchange in flexibility request | 66 |
| Appendix 3 Flex reservation mechanism | 67 |
| Appendix 4 Message transport mechanism | 69 |
| Appendix 5 Bibliography | 76 |

1 About UFTP

1.1 Purpose

The USEF Flex Trading protocol (UFTP) is a subset of the USEF framework. Focused specifically on the exchange of flexibility between Aggregators (AGRs) and DSOs, it describes the corresponding market interactions between them to resolve grid constraints by applying congestion management or grid-capacity management. UFTP can be used as a stand-alone protocol for flexibility forecasting, offering, ordering, and settlement processes. Its purpose is to support the use of flexibility markets for grid-congestion issues and standardize the set-up and operation of these markets across different DSOs.

1.2 Scope

Constraint management services

Constraint management services help grid operators (TSOs and DSOs) to optimize grid operation according to physical and market constraints. USEF [1] distinguishes four different flexibility services within the constraint management category: grid capacity management, congestion management, voltage control and controlled islanding. UFTP can be used for the first two of these.

Grid capacity management

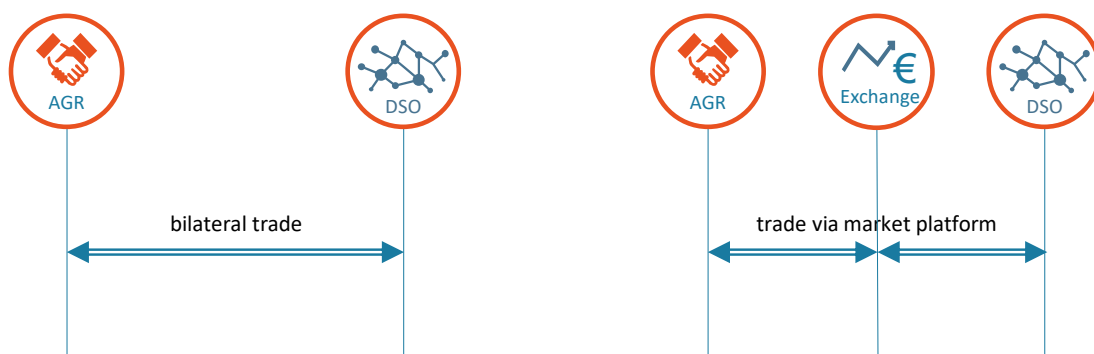
Grid capacity management refers to the use of explicit demand-side flexibility (by the DSO or TSO) to increase operational efficiency without impacting freedom of dispatch, trade and connect (copper plate principle). It is primarily used to defer grid reinforcements, optimize asset operational performance, reduce grid losses, or support planned maintenance. The implication is that AGRs (and prosumers) participate in the product on a voluntary basis; i.e. motivated by appropriate financial incentives.

Congestion management

Congestion management refers to the use of flexibility to reduce peak loads, in order to avoid system overload and the associated component or system failures. Contrary to grid capacity management, it will typically limit freedom of dispatch, trade and/or connect. It is primarily used in unanticipated overload situations; e.g. those which could not be forecast during long-term grid planning processes, or where load/ generation increases have outpaced grid reinforcement efforts. Congestion management is a temporary solution (typically until grid reinforcement takes place) and is a highly-regulated mechanism. At present, it is only available to TSOs in most European member states although it may be offered to DSOs in the future.

The two mechanisms differ in the freedom of dispatch, trade and/or connect and this is handled via contracts and will have impact on the settlement components.

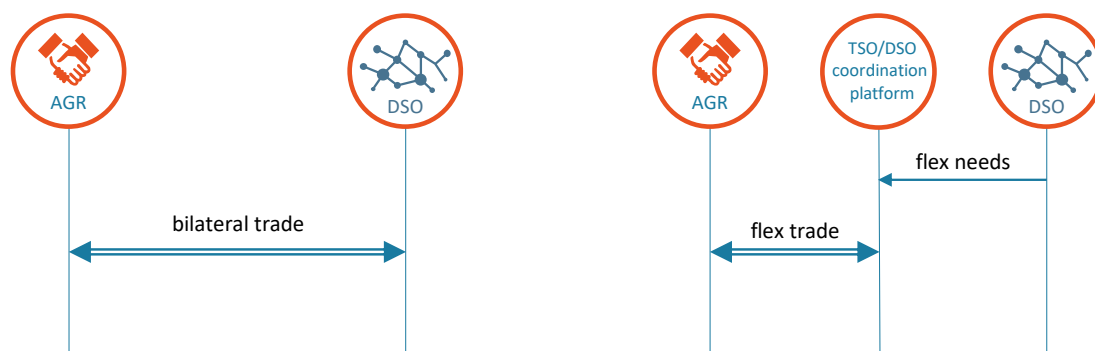
Bilateral trading and trading via a market platform



UFTP is designed for bilateral trading between the AGR and DSO. Trading via market platforms can bring added value, see [2]. In UFTP, market platforms may take the role of trading counterparty, hence serving as a DSO proxy in trades with the AGR, and as

an AGR proxy in trades with the DSO. When trading via a platform, the role of FlexRequests from DSOs to AGRs is less obvious; i.e., a DSO has free choice of the available FlexOffers. In addition, FlexOffers and FlexRequests do not need to be directly related and so UFTP supports unsolicited FlexOffers without requiring a reference to an underlying FlexRequest.

TSO/DSO coordination



USEF assumes that a DSO or TSO can freely and independently trade flexibility with AGRs. Any potential goal conflicts can be settled through the market mechanism. Alternatively, TSO and DSO requests can be aligned, resulting in a TSO/DSO coordination platform where TSOs and DSOs coordinate the tendering, trading, activation and/or settlement of flexibility for their own purposes (i.e. ancillary services). See [2] for a discussion on this topic.

UFTP can be applied without any further changes in a configuration where there is TSO/DSO coordination.

Alternative use

The text, graphs and examples in this document assume flex trading between AGR and DSO. However, UFTP can equally be used for grid constraint services between an AGR and TSO.

The messages defined in UFTP can also be used for flexibility trading in the plan phase between an AGR and BRP. See the USEF Framework Specifications [3] for a complete description of this alternative use.

1.3 Levels of compliancy

UFTP has mandatory components and optional components. The tables below list all core and optional components. Support of core components is mandatory for all implementations. Optional components are signaled throughout the document. Optional components in messages are always prefixed with [optional].

| Core UFTP components | USEF 2015 | UFTP | Signaled in CRO? |
|----------------------------------|-----------|------|------------------|
| UFTP Process: | | | |
| Day-ahead Flex trading | ○ | ○ | - |
| Redispatch responsibility choice | - | ○ | Y |
| Baseline choice | - | ○ | Y |
| Market messages: | | | |
| FlexRequest | ○ | ○ | - |
| FlexOffer | ○ | ○ | - |
| FlexOfferRevocation | ○ | ○ | - |
| FlexOrder | ○ | ○ | - |
| FlexSettlement | ○ | ○ | - |

| Optional UFTP components | USEF 2015 | UFTP | Signaled in CRO? |
|--------------------------------------|-----------|--------|------------------|
| UFTP Process: | | | |
| Intraday Flex Trading | ○ | ○ | N |
| Availability contracts (FlexOptions) | ~1 | ○ | N |
| Dynamic pooling | - | future | ? |
| Market messages: | | | |
| D-prognosis | ○ | ○ | Y |
| FlexReservationUpdate | - | ○ | N |
| FlexOffer: mutual exclusive offers | - | ○ | Y |
| FlexOffer: unsolicited offers | - | ○ | - |
| FlexOrder: partial activation | - | ○ | - |

1.4 Reading guideline

The USEF Flexibility Trading Protocol Specification is accompanied by 'USEF: The Framework explained' [4] which provides insight into the background, scope and concepts underpinning USEF. It is strongly advised that the USEF document is read in advance of this UFTP specification document.

USEF follows the principle of privacy & security by design and therefore the specifications are accompanied by a privacy & security guideline [5] which is applicable to the entire design, implementation and operation of USEF and derived products, services and solutions.

The UFTP specifications consist of three main elements:

- Chapter 2 – General description
Informative description about usage of the protocol and process description for USEF's market-based coordination mechanism
- Chapter 3 – Use case descriptions
Detailed descriptions of each process step of the process model, incl. scope & level, goal in context, preconditions, primary and secondary actors, the main scenario, etc.
- Chapter 4 – Message transport & descriptions
Syntax descriptions of all the messages required for the interfaces described in the framework, as well as detailed requirements for exchanging those messages. A process description, including the specifications and rules that apply to each process, are provided. A motivation for the design is included, wherever it is assumed beneficial.

Recommended Practice:

In order to make the universal descriptions of the framework applicable in practice and more comprehensible the specifications are accompanied by recommended practices. These paragraphs are clearly indicated, in blue, throughout the specifications.

¹ Availability contracts were mentioned in USEF2015, but not all details were worked out

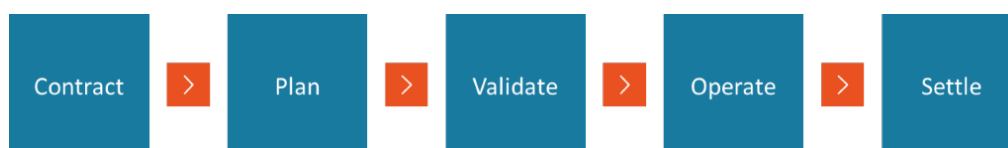
2 General description

This chapter describes the global information flows for flexibility trading between AGR and DSO for grid constraint purposes. In this respect, UFTP acts as a subset of USEF's market-based coordination mechanism (MCM), where AGR-DSO trading is an embedded part of the broader mechanism which applies to multiple flexibility services for all relevant market roles. In order to understand how UFTP is embedded in USEF, we have used some of USEF's general concepts to describe the UFTP processes. For further information on these, please see 'USEF: The Framework explained' [1]. Sections 2.2 to 2.6 describe the interactions in the different Phases of MCM: Contract, Plan, Validate, Operate and Settle.

2.1 USEF general concepts

USEF market-based coordination mechanism (MCM)

USEF's MCM facilitates the delivery of value propositions (i.e. marketable services) to various market parties without imposing limitations on the diversity and customization of the propositions. The USEF MCM is designed for all energy commodities and enables the market to optimize in time, capacity and power. The MCM provides access, under equal conditions, for all stakeholders to a single integrated market. This unique approach aims to deliver a future-proof market design. The USEF MCM operations scheme distinguishes five phases:



In UFTP, the general MCM phases are followed. However, processes in each phase are limited to the interactions between the AGR and DSO for flexibility trading. This includes interaction with the 'Common Reference':

- **Contract:** AGR and DSO negotiate FlexOptions. This is flexibility that is reserved for DSO purposes and can be invoked by the DSO when needed. Typically, a contract includes availability remunerations and activation remunerations.
- **Plan:** information exchange between DSO and AGRs related to congestion points. This information exchange through the 'Common Reference' involves communication with the Common Reference Operator (CRO). The AGR carries out an initial portfolio optimization.
- **Validate:** the DSO uses D-prognoses to validate whether the demand and supply of energy can be distributed safely without any limitations. If congestion occurs, the DSO can procure flexibility from AGRs to resolve grid capacity issues.
- **Operate:** in the operate phase, the actual assets and appliances are dispatched and the AGR adheres to its D-prognoses. When required, DSOs can invoke additional flexibility from AGRs to resolve unexpected congestion.
- **Settle:** in the settle phase, the flexibility that the AGR has sold to DSOs is settled. For this purpose, the actual consumed and produced volumes are allocated to the responsible parties first. Any unresolved or disputed volumes are reconciled shortly afterwards.

USEF Operating regimes

USEF recognizes four different operating regimes. The green regime is the classical grid without any limitations (copper plate) and the red regime is where power outages occur. USEF introduces two additional regimes. In the green and yellow regimes, the MCM assures optimal use of the flexibility available for BRPs (green and yellow) and DSOs (yellow). The orange regime is introduced as a fallback for situations where there is insufficient flexibility available for the DSO to avoid an outage—the DSO can limit connections to temporarily overrule the market.

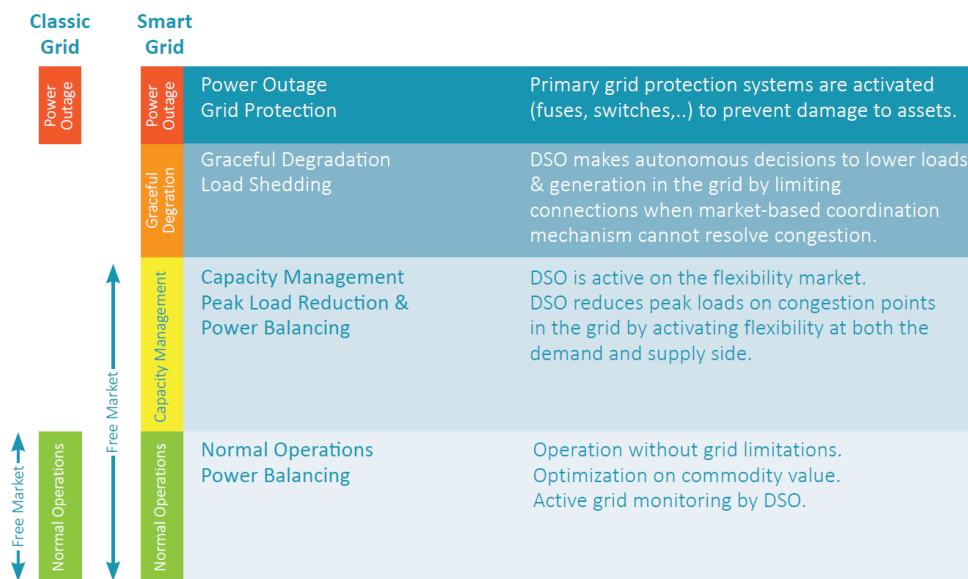


Figure 2-1: USEF Operating regimes.

UFTP flexibility trading takes place solely in the yellow regime. Full descriptions of all other regimes can be found in [3].

Day-ahead and intraday flexibility trading

The plan and validate phases aim to make optimal use of flexibility and maximize the freedom of dispatch and transactions of all stakeholders before the actual delivery of energy takes place. The timescale of these phases already ranges from years and months ahead, towards one day ahead and, ultimately, hours ahead before the actual delivery of energy (i.e. the operate phase) starts. This broad time window supports trading on various energy markets (i.e. forward market, day-ahead spot market and intraday spot-market) and the monitoring of changes in the required grid capacity.

USEF specifies that iterations between the plan and validate phases take place at least twice: initially, during the day ahead and, secondly, during intraday. Hence, two points in time are determined when the markets close: 1) end of day-ahead 2) end of intraday. This specification eases the iterative process of creating, aligning and converging plans and D-prognoses and fits well with many current national processes applied in the day ahead and intraday trading market. Please find more details in Section 2.4.5 (Iterations between the plan and validate phases).

UFTP supports both day-ahead and intraday trading. Intraday trading is optional for UFTP.

Congestion points

A congestion point is a set of connections which (directly) relate to a part of the grid where grid capacity might be exceeded because it may be insufficient to distribute the requested amount of energy; e.g. the secondary side of an LV transformer. Please note that a congestion point is a part of the grid where congestion *may* possibly occur, rather than *actually* occurs. It is the DSOs responsibility to determine congestion points.

Recommended practice:

How to determine which grid points are congestion points is the DSOs responsibility and therefore outside of the scope of USEF. However, USEF recommends declaring congestion points at the lowest possible level in the grid as this allows for detailed insight about local congestion while simultaneously, through aggregation, safeguarding the reliability of the grid safety analysis.

Congestion points are registered in the common reference, see 2.2.3 Common Reference.

Flexibility market time granularity

All flexibility exchange in USEF is based on a time granularity that is in line with the imbalance settlement period (ISP)², which is the time unit for which the imbalance of the balance responsible parties is calculated. For most EU countries this is a 15-minute interval, leading to 96 ISPs per day. Forecasts, flexibility offers, and orders and settlement are all based on ISPs. Flexibility is expressed in Power by a given Power unit which is to be interpreted as the *average* Power during an ISP.

Alternative baselines

All flexibility trading in USEF assumes that flexibility offers take the form of deviation from *baseline*, i.e. the default situation that would occur if no flexibility were activated. Typically, the baseline is derived via an AGR nomination referred to as the D-prognosis. However, UFTP allows for alternative baseline, where a D-prognosis is not required, and it is assumed that the AGR and DSO agree upon the baseline. An alternative baseline could be based on a measurement (MBMA-method), a mathematical formula or a reference group, etc. The choice of the alternative baseline and its establishment is out-of-scope for UFTP. The UFTP trading messages either refer to D-prognosis or to an external baseline reference. Throughout this document, the D-prognosis is used as the default baseline method. If UFTP is used with alternative baseline, then all references to the D-prognosis can be ignored.

2.2 Contract phase

2.2.1 Overall description

The contract phase is when DSOs and AGRs begin interaction. Typically, this involves a pre-qualification process, where the AGRs capability to deliver flexibility, portfolio and IT systems are assessed. AGR pre-qualification is out-of-scope for UFTP and therefore not further described in this document. See [6] for more information. In addition, the contract phase includes the exchange of information related to congestion points and associated connections through the common reference. Note that any contract negotiations for bilateral contracts also takes place in the contract phase (see below).

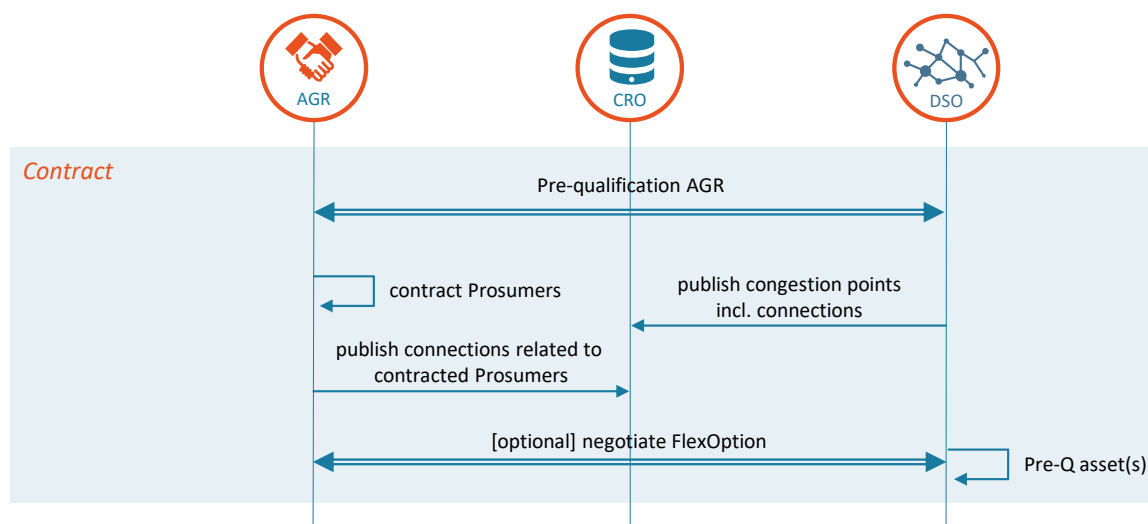


Figure 2-2: General information flow in the Contract phase

As part of its internal grid planning process, a DSO determines where congestion *may possibly* take place. Congestion points are declared whenever necessary, most likely several times a year, depending on the volume of trend analyses performed by a DSO and the condition of the grid. By publishing a particular congestion point, the DSO opens a flexibility market for it. AGRs publish the customers that they serve in the common reference. As a result, a DSO can determine whether there is liquidity on its market and AGRs know which of their customers can participate in a DSO's market.

² Commission Regulation (EU) 2017/2195 of 23 November 2017 establishing a guideline on electricity balancing defines an imbalance settlement period (ISP) as the time unit for which the imbalance of the balance responsible parties is calculated (Article 2(10)).

USEF introduces two alternative options for the DSO to procure flexibility from AGRs:

- **Long-term flexibility options:** activation of flexibility options in prearranged bilateral contracts. Contracts of this nature oblige the AGR to offer a fixed amount of flexibility to the market via daily FlexOffers. This product guarantees a certain regular availability of flexibility but the price (model) has to be arranged in advance. As a result, the DSO can select the contract(s) that provide the required flexibility at the lowest costs although these, typically, will not reflect all flexibility available in the market or the actual marginal costs for any flexibility supplied. This method is comparable to the way in which national TSOs contract balancing services and a tender or auction procedure can be implemented to manage it, alongside long-term flexibility options subject to prearranged specific conditions; e.g. which include a maximum or fixed price for the activation of the flexibility.
- **Short-term flexibility options:** procuring flexibility that AGRs have offered for a specific day. In this situation, the AGR has no contractual obligation to offer the flexibility to the market but decides to do so on a day-to-day basis. This product inherently deals with short-term flexibility which is only valid for a specific day. The price offered by AGRs will better reflect the marginal costs although the availability of the flexibility in this product is not guaranteed. This method is comparable with the way market parties are free to offer capacity to the TSO on the secondary and tertiary markets (regulating and reserve power) on a daily basis.

The first option seems attractive for the DSO as it guarantees a minimum supply of flexibility, therefore reducing availability risks in the validate phase, but this comes at a higher price. The DSO is responsible for finding the right balance between costs and certainty, while ensuring that grid reliability is always maintained. It is important to note that both products are part of the same flexibility market and therefore it provides natural growth options for the evolution of flexibility trading. While, initially, the availability of flexibility supply and demand might be a primary concern for both buyers and sellers, leading them to focus on long-term contracts, this will likely change as the market matures and availability becomes less of an issue. At that point, both buyers and sellers would be willing to have greater reliance on short-term flexibility which, in turn, would offer increased market liquidity.

2.2.2 Bilateral contract: FlexOption

For long-term flexibility, USEF introduces the **FlexOption**. This type of contract enables the DSO to take an option on future flexibility. In relation to this, USEF specifies that:

- a FlexOption is a bilateral contract between DSO and AGR to deliver a specified amount of flexibility, at a specific location (i.e. congestion point), for a specified time schedule and duration.
- FlexOptions force the AGR to send flexibility offers when the flexibility requests match the contract details.
- FlexOptions are tradable via secondary markets.

Recommended practice:

In order to make FlexOptions attractive to AGRs, a DSO could add remuneration components; e.g. availability remuneration and/or a minimum number of activations.

A FlexOption contract typically has the following ingredients:

| Element | Description | Example(s) |
|-------------------------------|--|--|
| ISPs and durations | The Imbalance Settlement Period (ISP) in which the flexibility should be delivered and the duration of the contract. | <ul style="list-style-type: none"> Every ISP between 17:00 and 19:00 on Fridays from December through March for the coming two years |
| Lead time | Time before the (recurring) flexibility option expires. | <ul style="list-style-type: none"> One week before dispatch The day before dispatch at 16:00 Does not expire |
| Congestion Point | DSO-specific: Congestion Point at which the flexibility should be realized. | <ul style="list-style-type: none"> |
| Amount | Amount of flexibility being offered, specified as a delta to a baseline value or drop to mechanism. | <ul style="list-style-type: none"> 10kW (average power per ISP) |
| Capacity remuneration | Price the AGR receives for offering the flexibility option | <ul style="list-style-type: none"> €20/MW/day €40/MW/ISP |
| Volume remuneration | Maximum or predefined price for the flexibility, to be paid when the flexibility is ordered by the DSO. | <ul style="list-style-type: none"> €20/MWh Less than €20/MWh €0 |
| Maximum number of activations | The Maximum number of activations at which the flexibility in the LT contract can be called upon. | <ul style="list-style-type: none"> Once per contract Once per month Unlimited |
| Recovery Time | Minimum time between activations | |
| Penalties | Amount the AGR is penalized for not offering the option of flexibility and/or the flexibility itself. | n/a |

UFTP does not include messages to communicate the contents of a FlexOption; this is assumed to be out-of-scope.

2.2.3 Common Reference

A DSO declares its congestion points in the contract phase. An AGR must be able to retrieve information about these points since this allows it to fulfil its obligation to provide the compulsory D-prognoses to DSOs for congestion points at which it represents customers. It also aids the AGRs portfolio optimization activities.

Since a prosumer can switch AGRs, and the congestion points in the grid change over time, USEF introduces a ‘common reference’. The common reference contains a list of connection identifiers (for example EANs) for each congestion point, as registered by the participating DSOs. Each AGR also registers the connections on which it represents prosumers. The common reference is shared between all involved parties, while respecting privacy and security principles [5] which ensures, amongst other things, that only necessary information is shared and therefore included in the common reference. The common reference is operated by the Common Reference Operator (CRO) role. Conceptually, it contains the entities and relationships shown in Figure 2-3:

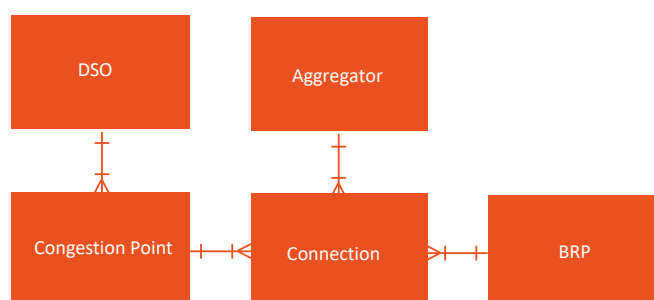


Figure 2-3: Logical structure of the Common Reference

The common reference is, typically, setup by the DSO and includes all of its congestion points. However, it could be extended to form a common reference at national or European grid level.

UFTP allows for the registering and tracking of changes in the common reference; e.g. a DSO might announce a new congestion point for the following week which is registered in the common reference. The history of all changes to the common reference is retained and can be retrieved for a certain date using the query function functions (see 4.2.8 - AGRPortfolioQuery and 4.2.10 - DSOPortfolioUpdate); this will also help resolve any potential disputes.

Accessibility of the data in the common reference

Access to the common reference data is limited according to the legal or contractual requirements of the various parties consulting it³. This implies that a DSO can only see the data for the connections it is responsible for. Consequently, it can see which AGRs are serving its connections, in order to ensure that it receives D-prognoses from them.

AGRs can only see information about the congestion points for which they serve connections. In addition, they can retrieve the number of connections per congestion point. Implementing the common reference this way ensures that the DSO does not need to share details of its grid topology.

The common reference can be operated in two modes: open mode and closed mode. If operating in open mode, the CRO will accept updates from any USEF-compliant participants. In closed mode, participants will need to be pre-configured in order for updates to be accepted.

2.3 Plan phase

USEF's plan phase aims to find an economically optimized program to supply the energy demand of both AGR and BRP portfolios for a certain period. In this phase, the AGR optimizes its portfolio and typically arbitrages between different flexibility markets to maximize the value of the available flexibility. The result of this process is a balanced portfolio. The exchange between AGR and BRP's and corresponding trades on the energy markets are out-of-scope for UFTP. The result, however, will lead to a D-prognosis in the validate phase.

The processes that take place during the plan phase are schematically depicted in figure 2-4.

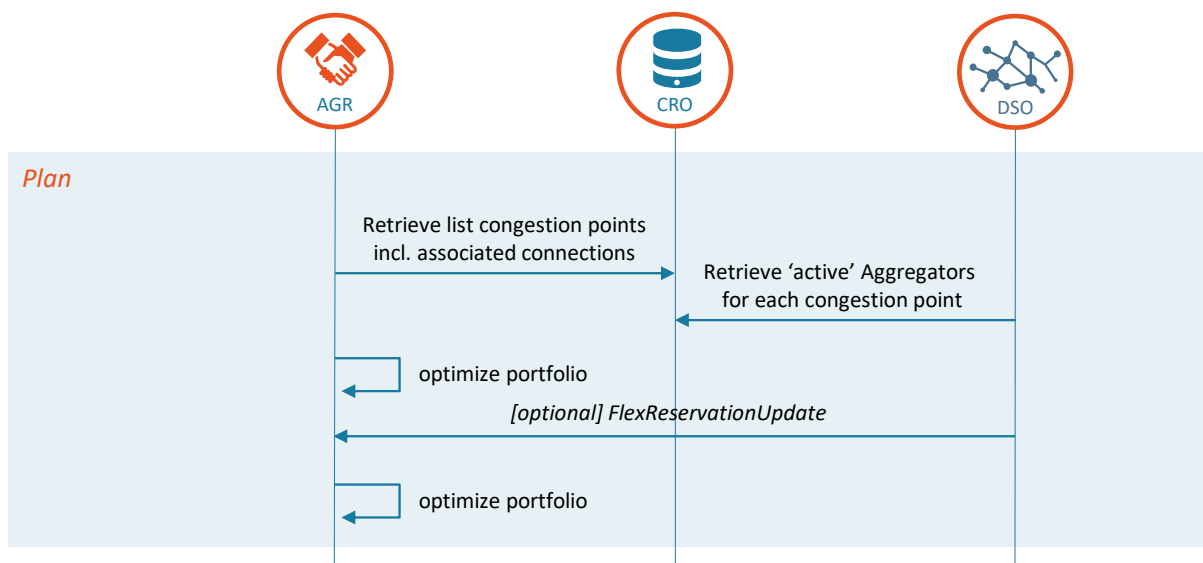


Figure 2-4: General information flow in the Plan phase

³ Another option is to treat the common reference as open data, therefore implying that everyone can access it. Although the open data model can ease the implementation of the common reference, the acceptance of such a model by prosumers is highly dependent on the culture in the geographical area in which the model is used.

Since the list of connections belonging to a congestion point, and the list of customers that are served by the AGR, may switch from day to day, USEF specifies that this information is requested on a day-to-day basis.

The AGRs portfolio optimization is out-of-scope and depicted only for reference. See [3] for a detailed description of this process. For UFTP, it is important to realize that the AGR optimizes its portfolio based on its clients' needs, optionally taking into account its long-term contractual obligations.

For bilateral contracts, the DSO might provide a FlexReservationUpdate message e.g. signaling which part of the contracted volume is still reserved and which part is not needed and may be used for other purposes. This will typically re-trigger the AGRs portfolio optimization process. More information about the flex reservation mechanism is added in Appendix 3.

2.4 Validate phase

2.4.1 Overall description

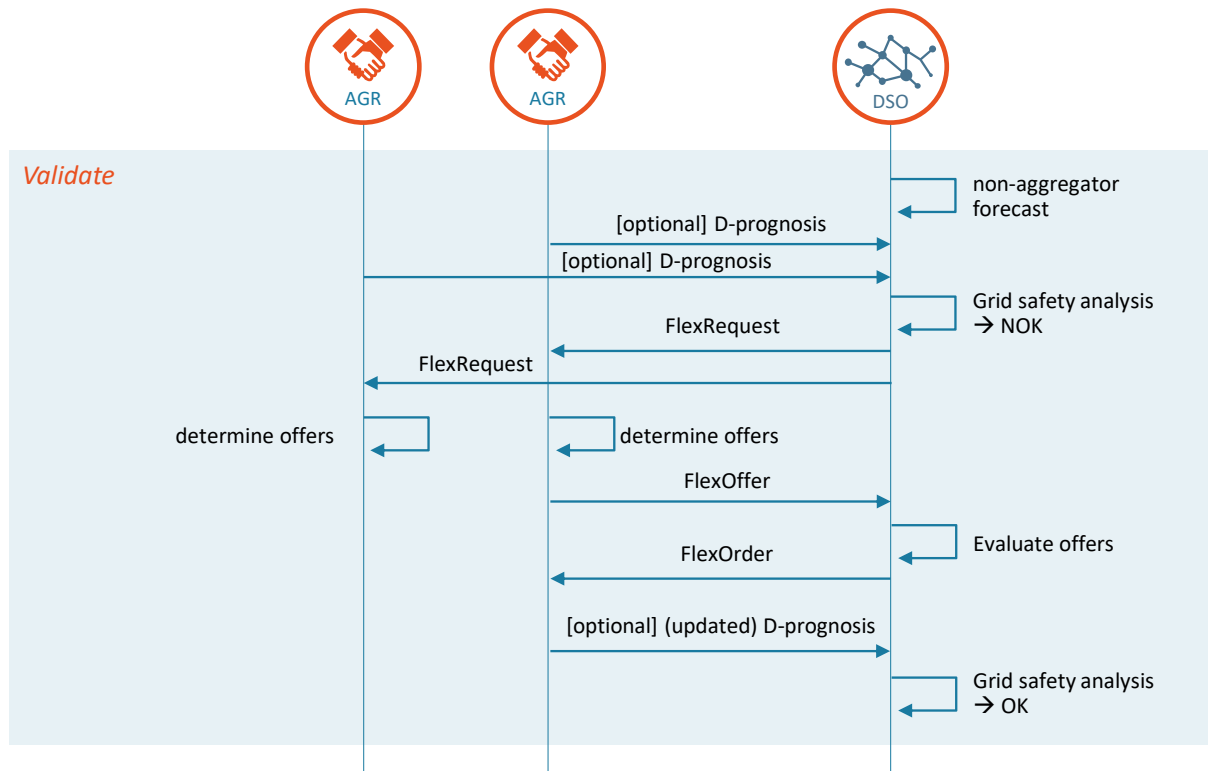


Figure 2-5: General information flow in Validate phase.

Assuming that D-prognoses are used as baseline⁴, each AGR creates a D-prognosis per declared congestion point and sends it to the DSO. Sending D-prognoses is not required when an alternative baseline method has been agreed. When all D-prognoses for a congestion point are received, the DSO combines these with the forecasts of non-AGR connections to execute a final grid safety analysis. This analysis determines whether it is possible to distribute the planned energy, or the limits of the distribution grid are reached. In the latter situation, USEF moves to the yellow regime and the DSO procures flexibility in the market to resolve these congestion issues. The process of flexibility trading is as follows:

1. The DSO requests all AGRs at the congestion point to provide flexibility. In this request, the DSO indicates the magnitude (amount of excess power) and timing (ISP) of the expected congestion, and how much capacity is available in the remaining ISPs. This is handled via a *FlexRequest*.
2. AGRs who are able and willing to change their load/generation profile in line with the request create *FlexOffers*.

⁴ UFTP allows for alternative baselines, see section 2.1. In case of alternative baselines, D-prognosis is optional.

3. If the offers are a good fit; i.e. help to mitigate congestion, the DSO can procure flexibility by placing a *FlexOrder*.
4. AGRs send an updated D-prognosis which incorporates the flexibility ordered.

If the flexibility offered is not sufficient to resolve the expected congestion, or no flexibility is offered, USEF moves to the orange regime. Please note that in this situation, USEF specifies that the DSO procures all offered flexibility from the market (i.e. from the yellow regime) to solve parts of the congestion. This is because any flexibility offered will still help to reduce the impact of the orange regime (e.g. the number of connections impacted by the graceful degradation). USEF's orange regime is outside the scope of UFTP, and, therefore, not further discussed in this document.

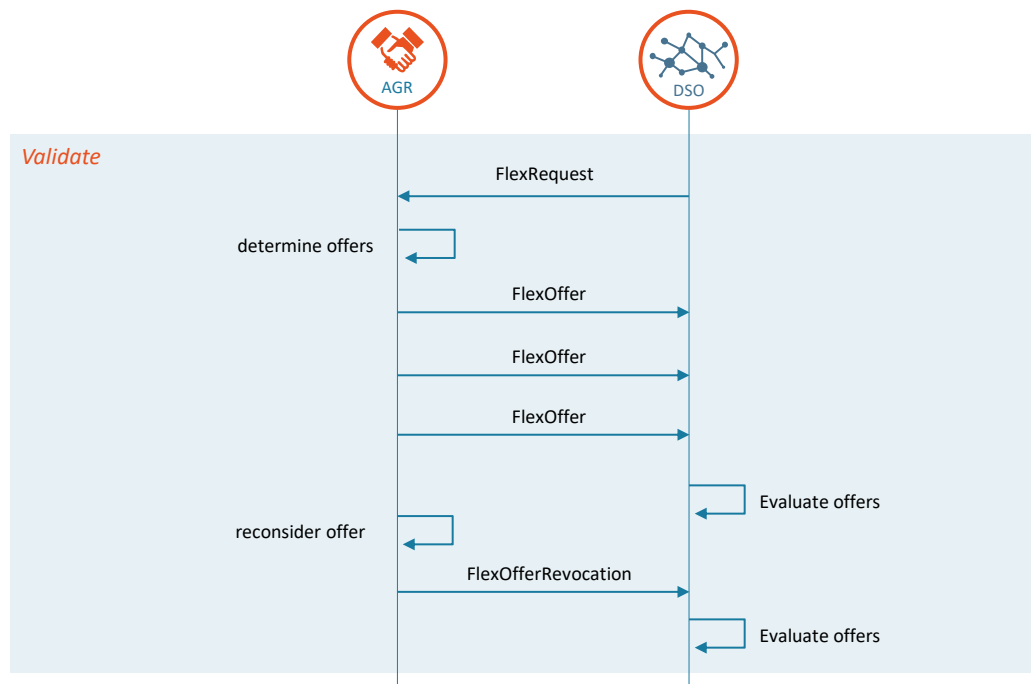


Figure 2-6: AGR may send multiple *FlexOffers* on the same request, and open *FlexOffers*, which are not yet ordered may be revoked by the AGR.

AGRs may respond with one or more *FlexOffers*. Where an AGR responds with multiple *FlexOffers*, the DSO can freely choose the most appropriate offer(s). Provided the flexibility has not been ordered via a *FlexOrder*, a *FlexOffer* may be revoked by the AGR (*FlexOfferRevocation*).

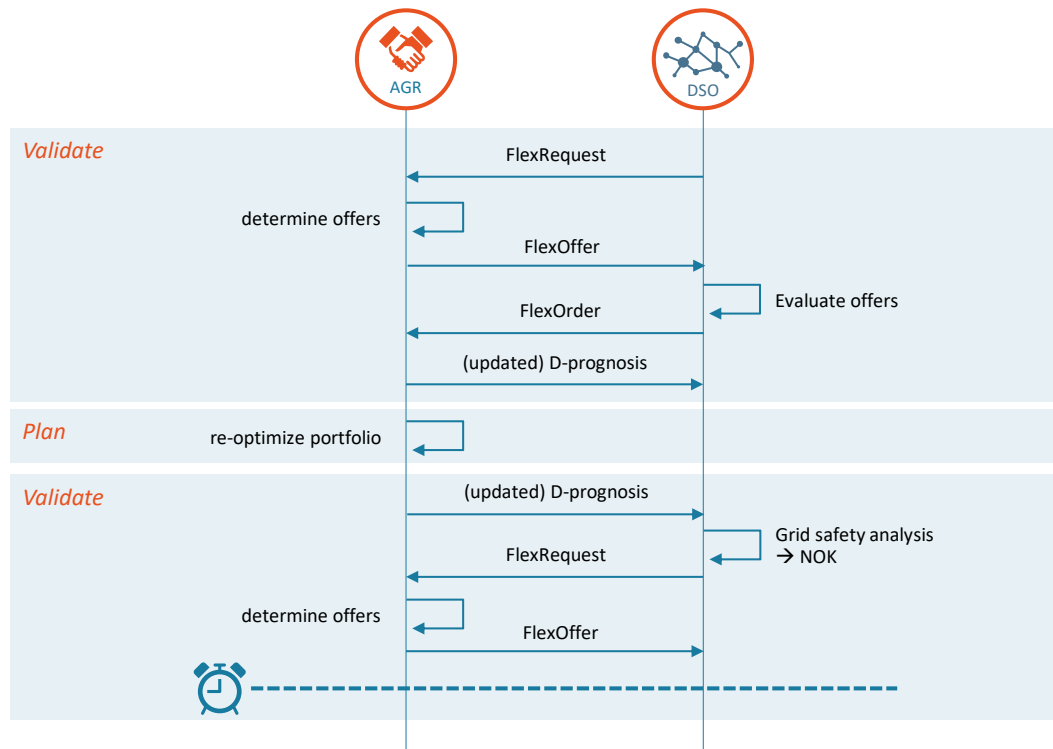


Figure 2-7: Flexibility trading may go on until gate closure time. This includes iterations between plan phase and validate phase. The last accepted D-prognosis before gate closure time serves as a basis for the next phase. Any open FlexOffers, which are not expired, may be ordered after gate closure.

2.4.2 Exchange of D-prognoses between AGR and DSO

When using D-prognoses as the baseline methodology, an AGR sends its D-prognosis per congestion point to the DSO. The DSO assesses the validity of each received D-prognosis and informs the AGR whether or not it is accepted. As long as the gate closure time has not passed, the AGR can update its D-prognoses and resend it to the DSO. During these iterations, an accepted D-prognosis⁵ is assumed actual, provided no associated updates are received and accepted by the DSO.

For the exchange of D-Prognoses, USEF specifies the following:

- An AGR is obliged to create a D-prognosis for its connections as soon as its connection point is declared by a DSO. The AGR is informed of this congestion point via the mandatory (daily) CRO queries.
- The D-prognosis only includes the forecasted load of those prosumers served by the AGR and that have a connection related to a congestion point.
- The D-Prognosis contains one full calendar day for the day-ahead process. In the intraday process, all remaining ISPs for that calendar day must be included
- The average power is given per ISP as a minimum time granularity
- The D-Prognosis does not include additional information; e.g. details about the available flexibility or prices

USEF provides the following recommended practices for the exchange of D-Prognoses:

- The D-Prognosis is sent day-ahead to the DSO; at the latest, two hours before the gate closure of the national balancing regime (day-ahead gate closure)⁶
- The DSO can perform a monthly evaluation of the accuracy and reliability of D-prognoses in order to motivate AGRs to improve the D-prognoses they supply

⁵ The accepted D-prognosis is the D-prognosis that used as a basis for the next period (Intraday) or phase (operate).

⁶ For the difference between day-ahead and intraday, please refer to Section 2.4.5

2.4.3 Processing D-prognoses

When all D-prognoses relating to a particular congestion point are received, the DSO creates a total prognosis for each congestion point. When accumulating the D-prognoses, the DSO must estimate and add the distribution requirements required for those connections which are not served by an AGR, and, therefore, not included in the D-prognoses. Using the accumulated D-prognosis, the DSO can determine whether it is possible to distribute the planned energy, or whether the limits of the distribution grid will be reached.

The timing of the FlexRequest is determined by the DSO, who must make a trade-off between waiting for all AGRs to send their D-prognoses⁷ before identifying grid constraints or issuing a FlexRequest based on its own grid forecasting process. It is the DSO's responsibility to combine the two sources of information. Where alternative baseline methods are used, the DSO will issue a FlexRequest based solely on its own information.

The grid safety analysis can lead to either of the following conclusions:

- That no congestion is expected. All power flows are within calculated safety margins.
- That congestion is expected as the energy flows exceed calculated safety margins. The DSO then starts the process of acquiring flexibility from the market.

Where no AGRs are active at the congestion point, there is no market to supply the DSO with flexibility to reduce its energy flows to levels below the safety margins. In this situation, the system moves to the orange regime. The resulting DSO processes are outside the scope of this document. See [4] for a global description.

2.4.4 Flexibility trading between the AGR and DSO

If the outcome of the grid safety analysis is that congestion will arise, the system is moved to the yellow regime and the DSO will need to take action. In order to solve the expected congestion, the following steps are taken:

1. The DSO requests all AGRs at the congestion point to provide flexibility. In its request, the DSO indicates the magnitude (amount of excess power) and timing (ISP) of the expected congestion and (optionally) how much capacity is available in the remaining ISPs
2. The AGRs receive the flexibility request for adjusting⁸ distribution requirements
3. AGRs generate offers for the flexibility. Note that these offers could either be the result of a voluntary offering process or compulsory, where there are prearranged bilateral contracts between DSO and AGR (FlexOptions)
4. The DSO receives the flexibility offers
5. The DSO procures flexibility to resolve the congestion issues by placing flexibility orders.
6. The DSO determines whether the expected congestion is solved using the ordered flexibility. If this is not the case, the system moves to the orange regime (see [4] for more details)
7. The AGRs receive the flexibility orders which result in the actual procurement of flexibility by the DSO.
8. AGRs provide an (updated) D-prognosis, incorporating the ordered flexibility. This step only applies when D-prognoses are used as baseline methodology.

When trading via a market platform AGRs typically offer their flexibility without an underlying request (unsolicited flex offers) and the DSO selects appropriate offers. Steps 1 and 2 are not applicable in this situation.

USEF specifies that, when sending a flexibility request, the DSO complies with the following:

- Information about expected congestion is provided to all registered AGRs;
- Expected congestion information includes the direction of the overload (production/consumption);
- Expected congestion information includes the amount of reduction needed;

⁷ USEF 2015 specified that the DSO should wait for all D-prognoses to be submitted. This would allow for a more precise determination of possible congestion and also ensure a fair process of offering and procuring flexibility. However, this constraint hinders the process in situations where there are missing or late D-prognoses.

⁸ Either by decreasing energy use, increasing local generation, or time-shifting load.

- Expected congestion information includes available grid capacity for other ISPs.

See appendix 2 for more details about this information exchange including rationale.

In the FlexRequest messages (see section 4.2.18), there is no single power value, rather a power space for each ISP, bound by two power values (“MinPower” and “MaxPower”). In addition, there is a distinction between ISPs with a ‘requested’ disposition – meaning that there is a request for a deviation on the power consumption/production – and ISPs with an ‘available’ disposition – meaning that there is available space to deviate on the power consumption/production.

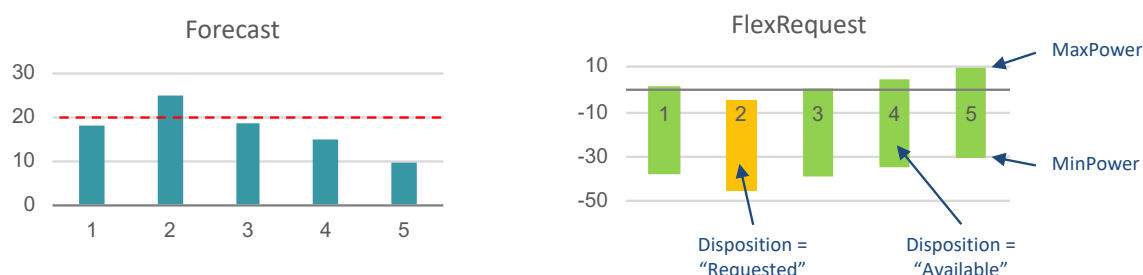


Figure 2-8: Example forecast for a Congestion point, leading to an example FlexRequest, both displayed in graphs

In the example used in Figure 2-8, congestion is expected at ISP 2 at a certain congestion point. The FlexRequest is transmitted to all AGRs that have prosumers on the congestion point, requesting a decrease of consumption (or an increase of production) on ISP 2. If AGRs choose to shift consumption from ISP 2 to another ISP, the available spaces in the ISPs from the FlexRequest indicate that there is more room for consumption in ISP 5 than in ISP 1, 3 and 4. A flexibility offer with a consumption shift to ISP 5 is therefore more likely to be ordered.

Note that a request is always relative to the agreed baseline. The essence of broadcasting the same request to all AGRs is that the requested deviation is the same for each AGR, relative to its own baseline. Decreasing power consumption is thereby equivalent to increasing power production and vice versa.

The distribution of the FlexRequest message is limited to AGRs which are registered via the CRO (see Section 2.3). The information on direction, amount and available capacity is provided in specific data structures that are part of the FlexRequest message (see Section 4.2.1).

For the procurement of flexibility by the DSOs, USEF specifies the following rules:

- A flexibility offer is valid until it expires or is revoked. It is the AGRs responsibility to determine the acceptance deadline. Revocation is not possible after the offer has been ordered.
- A flexibility order is definite and binding once it has been placed.
- The DSO chooses which flexibility offers it accepts to solve the congestion. In this regard, it is not obligatory for the DSO to start with the offer that has the lowest price. The DSO has the freedom to assess the balance between the price and quality⁹ of the flexibility offered in both long-term contracts and short-term offers.
- Bidding takes place at congestion point level, making every congestion point a local flexibility market.
- The DSO sends a flexibility request when the result of the grid safety analysis indicates possible congestion.
- A flexibility order is linked to a flexibility offer and a baseline for settlement purposes.

UFTP has two optional structures in the trading messages:

- [optional] A FlexOffer can contain multiple mutually exclusive options. The DSO is able to pick one of the options in its FlexOrder. Whether or not an AGR is allowed to add multiple options is signaled in the CRO.

⁹ For example, it is desirable for the DSO to procure flexibility that moves energy use from a moment in time of high load to a moment in time of very low load, rather than procuring flexibility that moves energy use from a moment in of high load to a moment in time at which the maximum of the distribution capacity is almost reached.

- [optional] AGRs may indicate the capability for partial activation in their offers. In all messages, a minimum activation percentage is provided i.e. all values lower than 100% mean that partial activation is possible. The DSO could use partial activation to better match the requested flexibility with available offers. Partial activation applies to all ISPs in the offer and also to the offered price. For example, if the DSO procures 80% of the offer, all ISPs in the FlexOrder should be 80% of the corresponding ISPs in the FlexOffer, and the settlement price is set to 80% of the original price in the offer. If the DSO has not implemented the option, it may simply ignore the structure.

Updates after flex has been ordered

USEF specifies that when ISPs are altered as a result of a flexibility deal, the resulting load for these ISPs is used for settlement. This means that deviation on ISPs with 'requested' disposition that have been traded, might lead to penalties for the associated AGR. Updates on other ISPs are always possible for the AGR but will require a new D-prognosis. Other AGRs can still change their D-prognoses for all ISPs without incurring a penalty. More details about the penalty calculation can be found in the settlement Section 2.6.

The load profile resulting from a flexibility deal will be stored independently from any subsequent updates of the D-prognosis, because it allows the DSO to verify the actual delivery of the flexibility for settlement purposes.

Recommended Practice for flexibility trading:

For the procurement of flexibility by the DSOs, USEF recommends the following:

- Publishing is in an anonymized way
- Pay-as-bid is used for pay-out.
- Timing of the flexibility orders is between 24 and 2 hours before delivery (day-ahead trading).

2.4.5 Iterations between the plan and validate phases

As a result of flexibility exchange with the DSO, the AGRs portfolio is not balanced and so it may wish to re-optimize it. By doing so, the MCM returns to the plan phase (see Section 2.3), the only difference being that the AGR now takes the flexibility it has already sold into account. In addition, during the plan phase, the AGR has the freedom to continuously re-optimize its portfolio whenever deemed necessary (e.g. when new and improved forecasts are available). The rationale behind this high degree of freedom for the AGRs is to ensure that the flexibility they offer is applied in the most optimal and cost-effective way (i.e. flexibility is used there where it creates most value).

However, these iterative processes also lead to the issue that D-prognoses change continuously, and continuous alignment is required between AGR/DSO. As a result, there is a risk that the iterative process between the plan and validate phases does not converge. In conclusion, there is a balance to be struck between making sure that the iterative processes converge, and offering a high degree of freedom to ensure that the available flexibility is used in the most optimal and cost-effective way.

Gate closure times

For reasons mentioned above, USEF specifies that there needs to be at least two moments in time when the iterations are forced to end. This moment is called the gate closure time.

In alignment with the current practices in liberalized energy markets, the two gate closure times are:

1. Day-ahead closure time: the moment at which the day-ahead market closes. During the day-ahead period, the plan and validate phases can be iterated as often as needed, as long as the system converges before the gate closure time. After the day-ahead closure time, the processes in the plan phase can only be restarted during the intraday period.
2. Intraday closure time¹⁰: the moment at which the intraday market closes. During the intraday period, the plan and validate phases can be iterated as often as needed, as long as the system converges before the gate closure time.

¹⁰ Note that intraday flexibility trading is optional in UFTP so intraday gate closure applies only when intraday trading is enabled by the DSO

After the intraday closure time, the processes in the plan phase cannot be restarted at all, as the MCM moves to the operate phase.

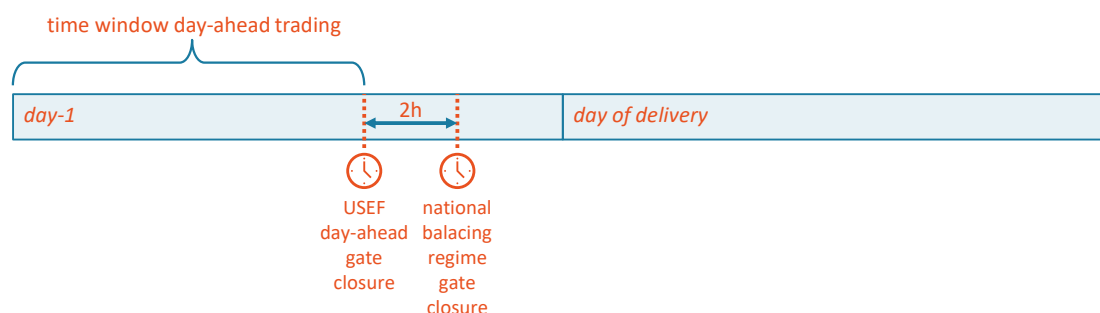


Figure 2-9: Example of USEF day-ahead gate closure time.

An example of the day-ahead gate closure is given in Figure 2-9 above. Please note that while there is a single day-ahead gate closure time per day, there will be multiple intra-day gate closures. This is illustrated in Figure 2-10. The ISP corresponding to the current time is in operate phase and ISP's more than one hour ahead are still in plan or validate phase.

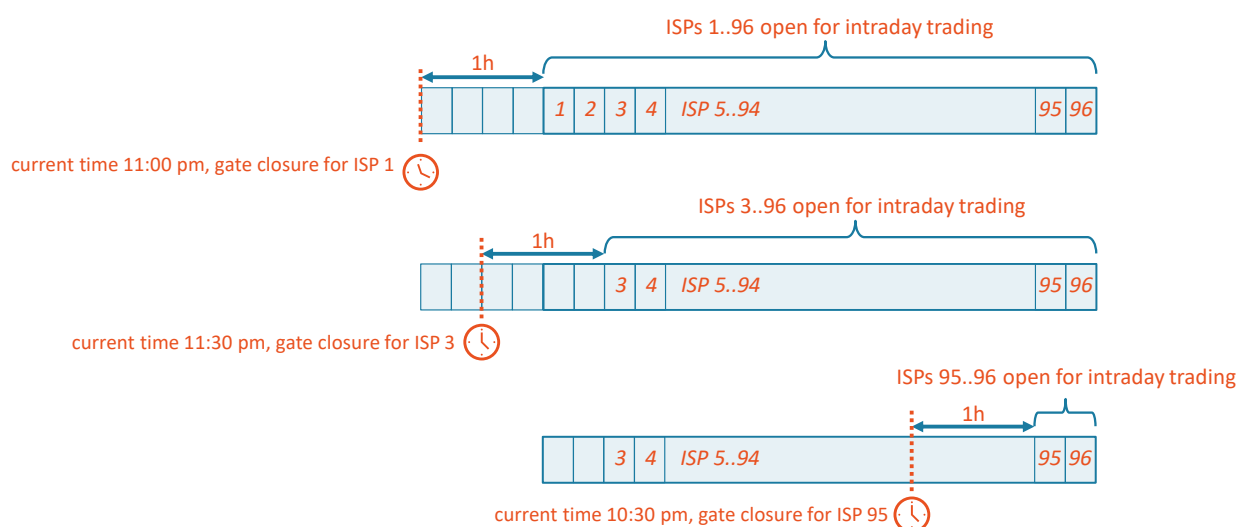


Figure 2-10: Example of USEF intra-day gate closure times.

Recommended Practice:

The national regulator determines the specific timing details of the gate closures. USEF recommends:

- A day-ahead gate closure time of 2 hours before the gate closure of the national balancing market
- An intraday gate closure time of one hour before the start of the ISP

2.4.6 Balance responsibility and redispatch options

The procurement of flexibility by the DSO may impact the AGRs balance position (nomination)¹¹. Therefore, USEF's validate phase is iterative with the plan phase, i.e. the AGR can make adjustments to its nomination via energy exchange with its associated BRP and/or energy trades on the energy markets. When the gate closure time is reached, all issues need to be resolved or there will be in an imbalance in the AGRs portfolio (which is the AGRs responsibility).

¹¹ This is not the case when the AGR can manage the deviation itself, e.g. one prosumer uses more energy, another prosumer uses less energy, and these two level each other out.

As an alternative, the procurement of flexibility is organized as an energy trade between AGR and DSO. In this situation, the AGR portfolio stays balanced and there is no need to iterate to the plan phase. Nevertheless, the AGR is free to re-optimize its portfolio at any time, for any reason, which could induce iterations. The DSO faces imbalance as a result of the trade and is responsible for this. The DSO's imbalance can be restored via a counter trade outside the congested area (redispatch).

USEF supports both ways of balance responsibility. The DSO determines the method of trading for a given congestion point and signals this via the CRO. See section 4.2.10 (DSOPortfolioUpdate)

Recommended practices for balance responsibility:

- USEF recommends that the AGR retains balance responsibility for day-ahead flexibility trading. Any imbalance risk as a result of DSO trades can be mitigated via day-ahead energy trades.
- USEF recommends shifting the balance responsibility to the DSO for intraday flexibility trading. This is equivalent to an energy trade from AGR to DSO. In order to neutralize its position, the DSO should then trigger a counter-trade (redispatch).

2.5 Operate phase

While (similar to the current liberalized energy market) the total energy system will remain in balance, without any congestion issues, as long as there are no deviations from the plans, it is unlikely that all plans will be executed exactly. Deviations can arise from a range of sources, ranging from changing weather expectations to an extension of a football match. Deviations can lead to: imbalances in supply and demand of energy at total system level (affecting the BRP); changes within the AGR portfolio (affecting the AGR); and local congestion in the distribution system (affecting the DSO). USEF's MCM is designed such that, during the operate phase, additional flexibility can be applied in order to compensate for these deviations. The processes for invoking this flexibility are depicted in Figure 2-11.

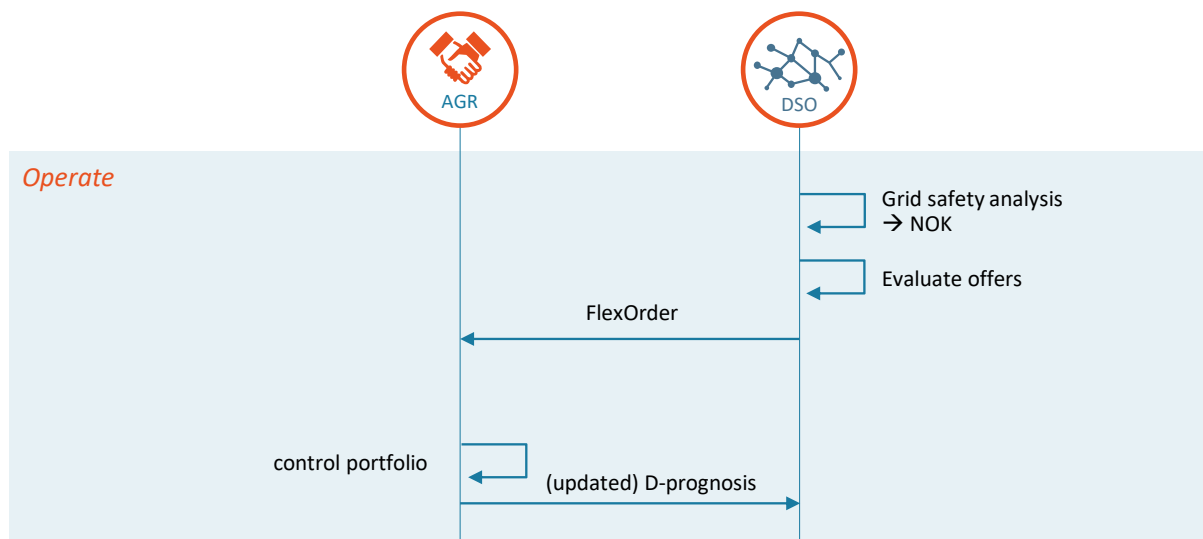


Figure 2-11: General information flows in the Operate phase.

During the operate phase, the AGR's main goal is to adhere to its plan and respect the D-prognoses.¹² In order to achieve this, the AGR must first plan to ensure assets operate in accordance with its forecast portfolio performance and that any flexibility sold is actually delivered.

¹² UFTP allows for alternative baselines, see section 2.1. In case of alternative baselines, D-prognosis is optional.

Next, the AGR measures the net demand or supply of its portfolio, in order to detect deviations from its plan and D-prognoses. Where deviations occur, the AGR re-optimizes its portfolio. It is possible that deviations could be solved using the flexibility contained within the portfolio itself.

If this is not the case, the AGR must change its plan (and probably limit its liabilities due to non-performance, to minimize fines) and control the assets to ensure that the new plans are realized and send an updated D-prognosis. If a FlexOffer has already been ordered, but the AGR is not able to comply, the AGR should inform the DSO as soon as possible through direct means outside of the UFTP protocol. During the operate phase, the AGR may wish to revoke outstanding (i.e. not ordered) FlexOffers if it is no longer able or willing to deliver the flexibility.

Although the DSO will reduce congestion risks in the validate phase, the DSO can still request that AGRs dispatch flexible power options to resolve potential grid problems in the operate phase. Again, due to time constraints, this will be done based on flexibility offers from the validate phase, which are still valid. When this flexibility is used, the corresponding BRP portfolio will no longer be in balance. Where the AGR is responsible for the balance, it will most likely charge the additional costs to the DSO. This is not considered further.

2.6 Settle phase

2.6.1 Overall description

In the settle phase, the AGR is remunerated for the delivered flexibility. There should be a check first, to ensure that the procured flexibility was actually delivered. A penalty may be applied when there is a mismatch between the baseline and the actual profile. Except for metering data, USEF assumes that both AGR and DSO have all the information to make these calculations. The DSO sends its calculation to the AGR for verification.

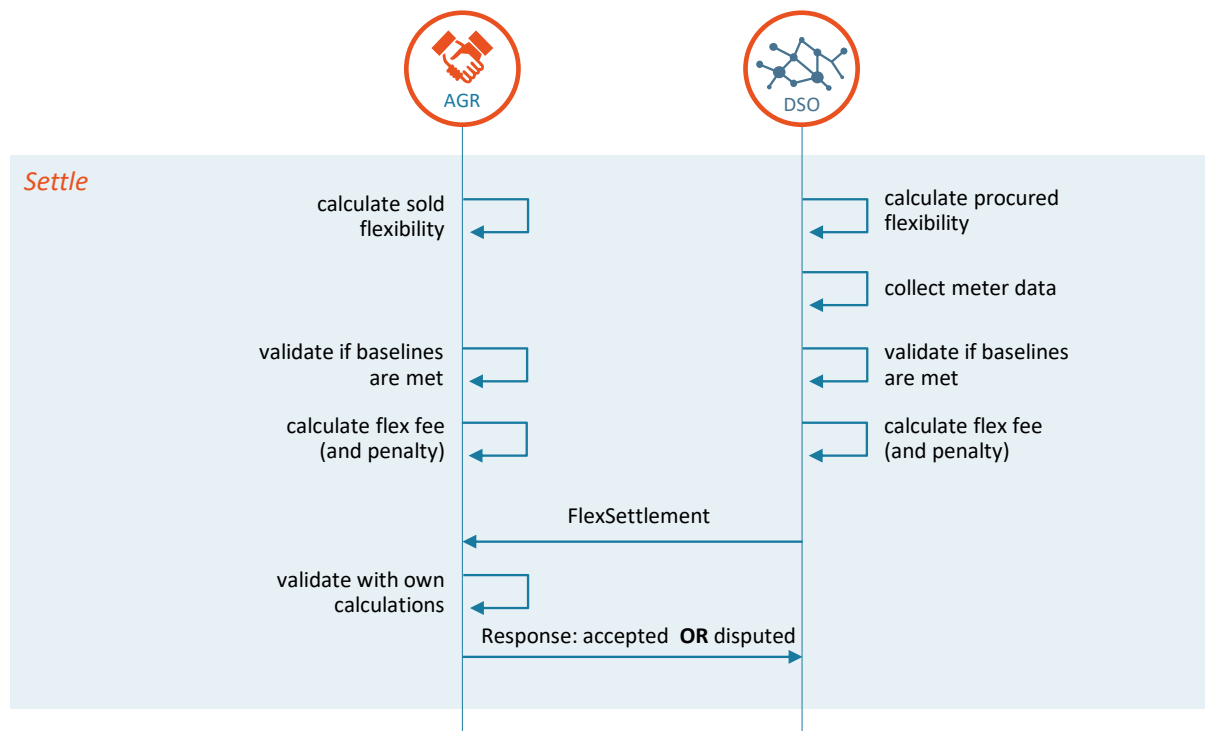


Figure 2-12: General information flow flex settlement between DSO and AGR

In order to perform the settlement, it is necessary for the DSO to compare the metering for each ISP against the baseline, as contained in the demand-prognosis message. In some regions the metering for the ISP periods for the fiscal meter may already be available to the DSO, through its own systems. However, where the DSO does not have access to the meter data, or where

plant metering is used for settlement assessment, the metering data must be communicated to the DSO by the AGR. For this purpose the AGR will use the Metering messages to send the daily metering per ISP from each connection point.

Recommended Practice: It is recommended to include the Power profile in the settlement messages.

Each month, the DSO will calculate the flexibility settlement volumes and prices for each AGR according to the steps below.

1. **Calculate procured flexibility**
For each congestion point and ISP, both the volumes and prices of the acquired flexibility are collected.
2. **Gather Metering Data**
This is done either based on the DSO's own metering records or using the metering message
3. **Validate if baselines are met**
This step is only performed where flexibility has been procured from the specific AGR, for a congestion point and ISP. The deviation between the final forecast (taking into account all procured flexibility) and the realized profile is calculated.
4. **Calculate flex settlement volumes and prices**
The settlement volumes and prices are calculated (see example below).
5. **Send settlement message to AGR**
This shows, per month, the accumulated settlement volume, accumulated deviation from the agreed baselines, and accumulated settlement price (accumulated over the ISPs and the congestion points).

The AGR will use its own information to do the same calculations

1. **Calculate sold flexibility** (equivalent to 1)
2. **Validate if baselines are met** (equivalent to 2)
3. **Plausibility check** (equivalent to 3)
The results are compared to the settlement message received from the DSO.

If the difference in the previous plausibility check is within limits, the AGR sends an acknowledgement to the DSO. If the difference exceeds predefined limits, the DSO's calculation is disputed. This is signaled via the SettlementResponseMessage. The dispute process is not part of USEF.

2.6.2 Settlement details

The DSO is responsible for settling the flexibility that it has acquired from the AGR. Within this settlement, the DSO needs to check whether the acquired flexibility has been delivered according to the agreements. Penalties can arise where agreements have not been met and this is considered an integral part of the settlement process.

This section describes how the supply of flexibility from the AGR to the DSO is settled. Several methods are possible, depending on how the flexibility is offered to the DSO and, more precisely, whether the DSO uses the mechanism of long-term and short-term flex options, and whether these flex options are rewarded, even if they are not exercised. Passive contribution to constraint management may also be rewarded.

Calculations are (in general) performed at ISP level, with settlement calculated over a one-month period.

Settlement components

The settlement consists of five components:

1. **Settlement of acquired flexibility**
The DSO will acquire flexibility through the market mechanism if congestion is expected, either ahead of time (validate phase) or real-time (operate phase). On average, the market price is determined by the equilibrium of demanded and offered flexibility (merit order). Following the recommendation given in Section 2.4.4, pay-as-bid pricing is used for individual settlements.
2. **Settlement of deviations from baselines**
AGRs that have sold flexibility to the DSO need to limit their capacity to the value stated in the baseline, corrected for

the sold flexibility. A penalty is raised for each ISP where the agreed capacity has (on average) been exceeded. The penalty is single sided, meaning that the AGR is allowed to deviate from its baseline as long as the deviation contributes to avoiding the grid constraint.

3. **[optional] Settlement of long-term flexibility options (bilateral contracts)**

The DSO may acquire long-term flexibility options to ensure that congestion can be avoided at all times. This may be necessary to justify a decision to delay or defer grid investments, or where grid constraints are expected in the (near) future. The DSO will determine (e.g. through audits) that the AGR is always capable of providing the contracted flexibility (at contracted times).

4. **[optional] Settlement of short-term flexibility options (FlexOffers which are not procured)**

When grid constraints are expected during the validate phase by the DSO, all AGRs are asked to place flexibility offers. AGRs whose flexibility bids are accepted by the DSO are expected to meet their (adjusted) baselines. However, it is just as important for the DSO that the other AGRs are also bound to their baselines. To enforce and reward this, the DSO may also accept an option on the flexibility offered by the AGR, even if the bid is not accepted, to ensure the AGR will not jeopardize grid constraints by invoking its flexibility for other purposes. The DSO will pay the AGR for the option, as this will put a constraint on the total capacity of the AGRs portfolio.

5. **[Optional] Settlement of passive contribution to constraint management**

Possible reward for AGRs that passively support the management of grid constraints – either by staying below the agreed value expressed in their baselines or by reducing more power than agreed in the exercised flexibility bid. This is specifically relevant if constraints have been resolved by passive contributions, where other AGRs have exceeded the agreed capacities. In this situation, the passive contribution can be rewarded using the raised penalties (zero sum calculations).

The settlement method implicitly assumes that each AGR has the motivation and capacity to produce high quality baselines.

Settlement calculations

(numbers refer to the settlement components above)

| Nr. | Amount | P = | Q = | Remark(s) |
|-----|--------------|--|---|--|
| 1 | $P \times Q$ | P = flexibility fee | Q = power called (“Delivered Flex Quantity”) | The flexibility fee is determined by the equilibrium of demanded and offered flexibility (flexibility market). |
| 2 | $P \times Q$ | P = excess penalty (asymmetrical, only applied when deviation has contributed to congestion) | Q = allocation – initial baseline – Σ FlexOrders (“Power Deficiency Quantity”), where <i>allocation</i> is the average realized power during the ISP | This penalty only applies to AGRs that have sold flexibility (for this ISP), and corresponds with not meeting the agreements of the flexibility bid. |
| 3 | negotiated | n.a. | n.a. | The DSO will pay compensation for the long-term flexibility contract in proportion to the agreed power, the duration of the time slot, the nomination lead-time and the duration of the contract. Fixed fee may be derived from: <ul style="list-style-type: none"> ■ The costs for allowing thermal limitations to be violated; ■ The avoided costs for grid reinforcements; ■ The value of flexibility for alternative uses; ■ The costs of alternative flexibility options. |

| | | | | |
|---|-------|---|--|--|
| 4 | P x Q | P = flexibility option fee | Q = power option | In this settlement component, an AGR can be rewarded for participating in the flexibility market through short-term products. Rationale for relating the flexibility option settlement to the capacity offered is that an option to cap the capacity of an AGR with twice the capacity of another AGR, has twice the value to the DSO. |
| 5 | P x Q | P = contribution fee (asymmetrical, only applied when deviation has relieved congestion). Derived from excess penalties raised. | Q = allocation – baselines – power flexibility called during Operate phase where <i>allocation</i> is the average realized power during the ISP. | |

Settlement example

For a specific ISP we assume:

- The AGR has specified a total load of 10 MW in its initial baseline
- During one or more flexibility requests, a total of 2 MW of flexibility has been acquired by the DSO, yielding an adjusted baseline of 8 MW
- The table below shows how different realizations are settled. The realization is specified by the allocation, representing the average power during one specific ISP for one congestion point. In this example the following (illustrative) prices are applied:
 - Flex price equals 7 €/ MW
 - Baseline deviation penalty equals 11 €/ MW (single sided)

| Allocation (MW) | Flex realized (MW) | Delivered flex quantity (MW) | Flex paid (€) | baseline deviation (MW) | Power deficiency quantity (MW) | Penalty raised (€) | Settlement (€) |
|-----------------|--------------------|------------------------------|---------------|-------------------------|--------------------------------|--------------------|----------------|
| 7 | 3 | 2 | 14 | -1 | 0 | 0 | 14 |
| 8 | 2 | 2 | 14 | 0 | 0 | 0 | 14 |
| 9 | 1 | 1 | 7 | 1 | 1 | -11 | -4 |
| 10 | 0 | 0 | 0 | 2 | 2 | -22 | -22 |
| 11 | -1 | 0 | 0 | 3 | 3 | -33 | -33 |

Elaboration of the calculations performed in the table:

1. *Allocation* shows possible results of the realization of the AGR portfolio
2. The *flex realized* equals the baseline associated with the flex offer minus the allocation
3. The *delivered flex quantity*, represents the flexibility that is both acquired (therefore maximum of 2) and delivered. Passive contributions (as in the row with allocation=7) are not rewarded.
4. The *flex paid* equals the *delivered flex quantity* times the flex price
5. The *baseline deviation* equals the *allocation* minus the adjusted baseline
6. The *power deficiency quantity* equals the *baseline deviation*, where negative deviations are set to 0 since these are not penalized.
7. The *penalty raised* equals the *power deficiency quantity* times the *penalty*
8. The settlement price equals the sum of the *flex paid* and *penalties raised*.
9. The monthly totals of the elements: *delivered flex quantity*, *power deficiency quantity* and *Settlement* are sent to the AGR.

3 Use case descriptions

3.1 Scope

This chapter provides details of the functions that contain interactions relevant to the DSO and AGR, in order to be active in the USEF MCM. To be USEF compliant, these functions must be implemented according to the use case descriptions in this chapter.

The use cases have been derived from the MCM sequence diagrams detailed in Chapter 2. Use cases are separate activities that require interaction between DSO and AGR, each containing multiple messages in both directions. DSO and AGR internal processes are not addressed in this chapter.

The messages types that occur in these use cases are mentioned and further described in Section 4.2.

3.1.1 Introduction

The core of the USEF specification is the MCM and the processes governing it. These define how the different stakeholders active in the MCM should behave and interact. All use cases related to these functions are described to the extent relevant for the USEF specifications. All aspects not specified by USEF are explicitly mentioned. An example of this is that USEF does not specify how an AGR optimizes its portfolio or determines how much flexibility is available, where or when. The result of this portfolio optimization is, however, a prerequisite for participating in the flexibility bidding process.

For the flexibility market to function, it is essential to have a functioning common reference populated with all AGR-represented connections.

Given the open nature of the USEF specifications, any implementation can choose to adopt all or only part of the specifications. In any situation, all participants within a single USEF market will have to adopt a common specification of the market processes and information exchange.

3.2 Privacy and security by design

USEF – like most complex information systems – deals with sensitive data and therefore requires security and privacy preservation measures. Privacy & security are system-wide issues; protection of individual subsystems/components is not enough; the system is as strong as the weakest link. USEF therefore follows the principle of privacy & security by design.

USEF provides a separate privacy and security guideline [5] that lists approximately 50 design principles which, combined, present a complete view of the privacy and security aspects associated with smart energy systems. It takes into account the current legal and social views on privacy & security and links these to the future directions in which they will likely evolve.

The guideline forms the basis for the logical security architecture that is reflected in USEF's process flows and use cases. Some remarks about security and responsibility:

- Personal and personally identifiable data represent commercially sensitive information. These include:
 - connection identifiers
 - existence of congestion points
 - details on AGRs associated to a connection
 - portfolio data
 - forecast data, content of D-prognoses and the fact whether or not a given AGR submitted a D-prognosis in a timely fashion
 - the existence, content and timing of a flexibility request and other flexibility trading messages
 - settlement items, including the acknowledgement or disputation of their correctness

- Personal and personally identifiable data (such as forecasts or smart meter readings) should not be retained longer than strictly required. The retention period should be specified and be communicated to the data subject.
- The exact identity of the AGR for each connection should not be disclosed to the DSO. Instead, only aggregate connection counts should be provided (which represent commercially sensitive information).
- It is the responsibility of the CRO to have policies regarding access, data retention, data security and conflict resolution compliant with the USEF privacy & security guidelines.
- The CRO should only disclose the existence of the congestion point on a need-to-know basis, i.e. only to those AGRs which provide an associated connection identifier in their queries.

3.3 Common concepts

The USEF specification standardizes the logical interfaces and defines the minimum functionality of the components in the form of use cases (this chapter) and message transport specifications and message descriptions (Chapter 4). This leaves room for innovation and possibilities to develop specific implementation architectures based on e.g. size of the market, specific local circumstances or commercial exploitation of USEF platforms. Stakeholders involved in a USEF market can and must develop business functions and capabilities independently, and focus on their core business and competitive advantage.

While the USEF specification is technology and implementation agnostic, there are some basic principles to which any USEF implementation architecture must adhere. Some of these are listed as high-level requirements. When an ICT implementation architecture is defined, the actual implementation requirements must be defined and included in the design.

- A USEF market implementation will typically consist of multiple information systems interacting together according to the USEF interaction standard in order to run the market processes. The USEF foundation does not want to narrow down the open nature of the USEF specifications by defining exactly how the information system architecture must be implemented.
- The USEF specifications do not currently define exceptions and how to handle them. Therefore, the use cases also prescribe only the default main scenario without error handling, and only the successful outcome. The single exception to this rule is described in the next bullet. USEF does provide some recommended practices for handling some of the failure handling.
- Any USEF IT architecture must adhere to the USEF privacy & security guideline (see [5]). In the USEF use cases the most relevant privacy & security considerations are provided.

3.4 Reading guidelines

This chapter contains separate sections per USEF phase: Plan, Validate, Operate, and Settle. In each section the use cases are specified in a standard table format:

Table 3-1: Standard table format for USEF use cases.

| | |
|---------------------------|--|
| Goal in context | High-level description of the activity. |
| Preconditions | The state of the system that must be present prior to a use case being performed. |
| Successful outcome | The state the system will be in immediately after a use case has finished without failure. |
| Failure outcome | The state the system will be in immediately after a use case has finished after failure, plus identification of the failure. |

3.5 Message validation

Incoming messages need to be validated by the recipient for their correctness. Validation of the semantics is performed by the HTTP over TLS (explained in Appendix 4 Message transport mechanism), but the content still needs to be checked for possible discrepancies. If the message passes validation, a response message is sent with 'accepted' in the result field. If it fails validation, a response message is sent with 'rejected' in the result field, along with a description of the reason why the message

is rejected in the RejectionReason field. Some causes to reject a message are specific for the type of message and are mentioned in the remainder of this chapter. Other causes are more generic and are listed in the table below.

This list is not all-encompassing: stakeholders are allowed to formulate other reasonable causes to reject a message, as long as it is clearly explained in the RejectionReason field. This field can contain multiple reasons for rejections (separated by a semicolon).

| RejectionReason | Cause of rejection |
|----------------------------------|---|
| Invalid Sender | Despite being schema-compliant, the syntax, type or semantics of the message were unacceptable for the receiving implementation. |
| Invalid Message | There is a mismatch between the SenderDomain/Role combination in the message wrapper and the inner XML message. |
| Unknown Recipient | The RecipientDomain and/or RecipientRole specified in the inner XML message is not handled by this endpoint. |
| Barred Sender | This endpoint is explicitly blocking messages from this sender. |
| Duplicate Identifier | The MessageID attribute of the inner XML message is not unique, and has already been used for a message with different content. This message has been rejected. |
| Already Submitted | The MessageID attribute of the inner XML message is not unique, but since the message content is the same as that of a previously accepted message, this copy can be considered to be successfully submitted as well. |
| ISP duration rejected | The message specifies a ISP duration that is not the agreed-upon common value for the market in which it is used. |
| Time zone rejected | The message specifies a time zone that has a different UTC offset than is the agreed-upon common value for the market. |
| Invalid congestion point | Unknown congestion point or the recipient is not active at this congestion point. |
| Unknown reference <Reference> | The message with the sequence where is referred to is unknown. For <Reference> the concerning reference field name can be filled in (for example FlexRequestSequence or PrognosisSequence). |
| Reference Period mismatch | The message(s) with the sequence where is referred to contains a different Period. |
| Reference message expired | The message that is referred to is expired. |
| Reference message revoked | The message that is referred to is revoked. |
| ISPs out of bounds | One or more ISPs are outside the tolerated boundaries: ISPs do not exist. |
| ISP conflict | One or more ISPs are defined more than once, possibly because of an incorrect duration. |
| Period out of bounds | Period of the message is inappropriate. For example: a FlexRequest with a Period in the past or a settlement item in a FlexSettlement with a Period outside the concerning settlement period. |
| ExpirationDateTime out of bounds | ExpirationDateTime is in the past or exceeds the ISPs in the message. |

3.6 Use Cases – Contract phase

In section 2.2 the informative description of the contract phase processes has been given. In this chapter use cases will be described as derived from the contract phase.

The USEF MCM contract phase specifies the following use cases:

Table 3-2: Use cases for the Contract phase.

| Name | Direction | Message types |
|--|-----------|---|
| Publish Congestion Points (Long-term) | DSO → CRO | DSOPortfolioUpdate / DSOPortfolioUpdateResponse |
| Publish Connections | AGR → CRO | AGRPortfolioUpdate / AGRPortfolioUpdateResponse |

3.6.1 Publish Congestion Points (Long-term)

Once a flexibility market is functional for an area, the DSO has to update and publish a register of possible congestion points, including the connection identifiers of connected prosumers. This register is maintained by the Common Reference Operator. If operating in open mode, the CRO will accept updates from any USEF-compliant participants implementing the DSO role. In closed mode, participants will need to be pre-configured in order for updates to be accepted.

Once populated, the common reference will allow AGRs to determine whether there are prosumers in their portfolio that can offer flexibility, via their connections, to one or more congestion points.

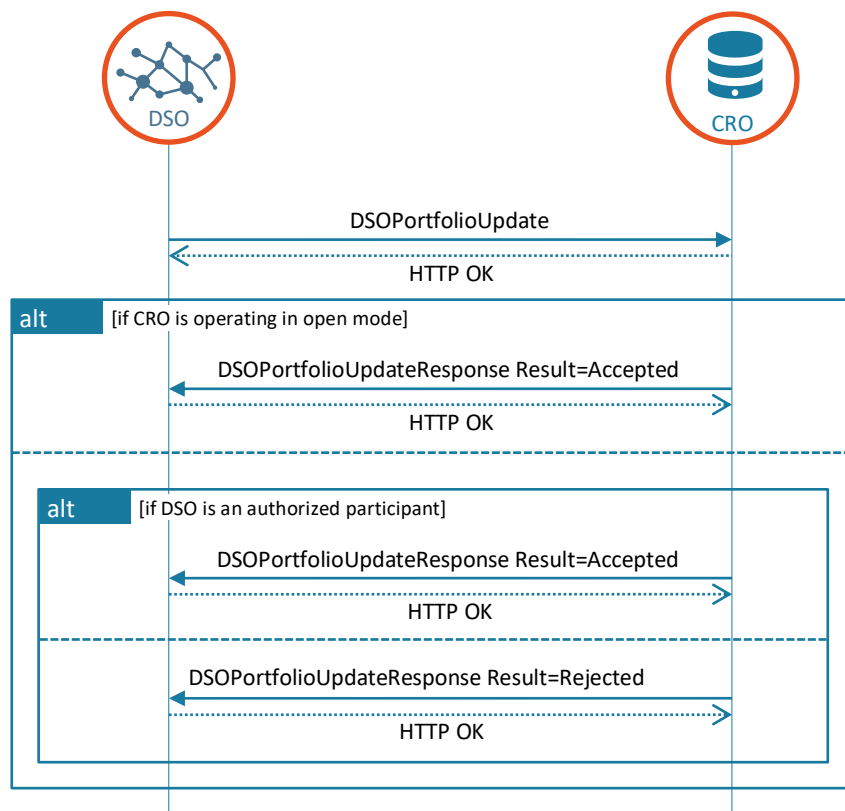


Figure 3-1: Exchange of congestion points including connections by DSO

| Publish Congestion Points / DSO Portfolio | | |
|---|--|---------------------------|
| Goal in context | In order to predict congestion and create a flex market: publish Congestion Points – including their associated Connections – to the Common Reference, so they become available to the AGRs. | |
| Preconditions | Common Reference is available to DSO and AGRs | |
| Successful outcome | The Common Reference is updated with the latest Congestion points and associated Connections (i.e. all connected Prosumers) | |
| Failure outcome | RejectionReason | Cause of rejection |

| | | |
|--|--|---|
| | <See section 3.5> | DSOPortfolioUpdate failed to pass validation by the CRO |
| | Unauthorized | CRO is operating in closed mode and the DSO is not pre-registered as an authorized participant |
| | Connection conflict: [<Connection EntityAddress>] at [<CongestionPoint EntityAddress>] | A connection is transmitted before at another Congestion Point. Return EntityAddress of the concerning Connection and Congestion Point where it has been placed before. |
| | Subordinate sequence number | The message sequence is lower than that of a previously received DSOPortfolioUpdate |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

It is the responsibility of the CRO to have policies regarding access, data retention, data security and conflict resolution compliant with the USEF privacy & security guidelines.

If the CRO detects strange behavior in the periods (e.g. an EndPeriod in the far past, an EndPeriod earlier than the StartPeriod or an EndPeriod of a congestion point earlier than the EndPeriod of a connection that it contains), it is up to the CRO to decide whether to reject the message, adjust the period or reply with a warning.

A DSO can update its portfolio by retransmitting the DSOPortfolioUpdate. It is essential to use a sequence number that is incremented each time a new revision is sent so the order of transmission can be traced.

To remove a congestion point, the DSO can transmit a new DSOPortfolioUpdate where the EndPeriod is the current time. The congestion point is then immediately expired. It is not necessary to add connections to this message since they will also expire.

3.6.2 Publish Connections

Once a flexibility market is functional for an area, the AGR has to publish a list of the connection identifiers of the prosumers it has contracted. This list is stored and, subject to access controls, made available to other market participants by the CRO.

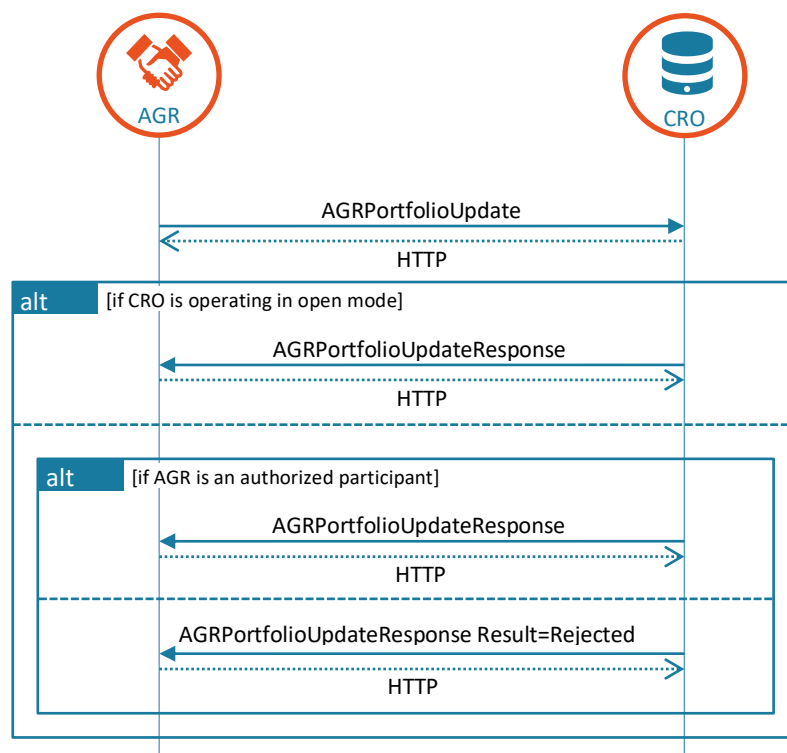


Figure 3-2: Exchange of connections by AGR

| Publish Connections / AGR Portfolio | | |
|-------------------------------------|---|--|
| Goal in context | Publish all contracted Prosumer Connections in the Common Reference, in order to later discover Congestion Points and associated DSOs and to register the AGR's presence at Connections to the associated DSOs. | |
| Preconditions | Common Reference is available to DSO and AGRs | |
| Successful outcome | The Common Reference is updated with the all Prosumers (listed by their Connection identifiers) represented by the AGR. | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | AGRPortfolioUpdate failed to pass validation by the CRO |
| | Unauthorized | CRO is operating in closed mode and the AGR is not pre-registered as an authorized participant |
| | Subordinate sequence number | The message sequence is lower than that of a previously received DSOPortfolioUpdate |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

It is possible to transmit connection points to the CRO that have not yet been declared by the DSO.

An AGR can update its portfolio by retransmitting the AGRPortfolioUpdate. It is essential to use a sequence number that is incremented each time a new revision is sent, so the order of transmission can be traced.

3.7 Use Cases – Plan phase

In section 2.3 the informative description of the plan phase processes are provided. In this chapter, use cases will be described as derived from the plan phase.

The USEF MCM plan phase specifies the following use cases:

Table 3-3: Use cases for the Plan phase.

| Name | Direction | Message types |
|--|-----------|---|
| Retrieve Congestion Points | AGR → CRO | AGRPortfolioQuery / AGRPortfolioQueryResponse |
| Retrieve Active Aggregators | DSO → CRO | DSOPortfolioQuery / DSOPortfolioQueryResponse |
| Exchange Flexibility Reservation Update | AGR ← DSO | FlexReservationUpdate / FlexReservationUpdateResponse |

The use cases are explained in the following sections.

If operating in open mode, the CRO will accept updates from any USEF-compliant participants implementing the AGR role. In closed mode, participants will need to be pre-configured in order for updates to be accepted.

3.7.1 Retrieve Congestion Points

The common reference allows an AGR to determine whether there are any congestion points where prosumers in its portfolio can offer flexibility.

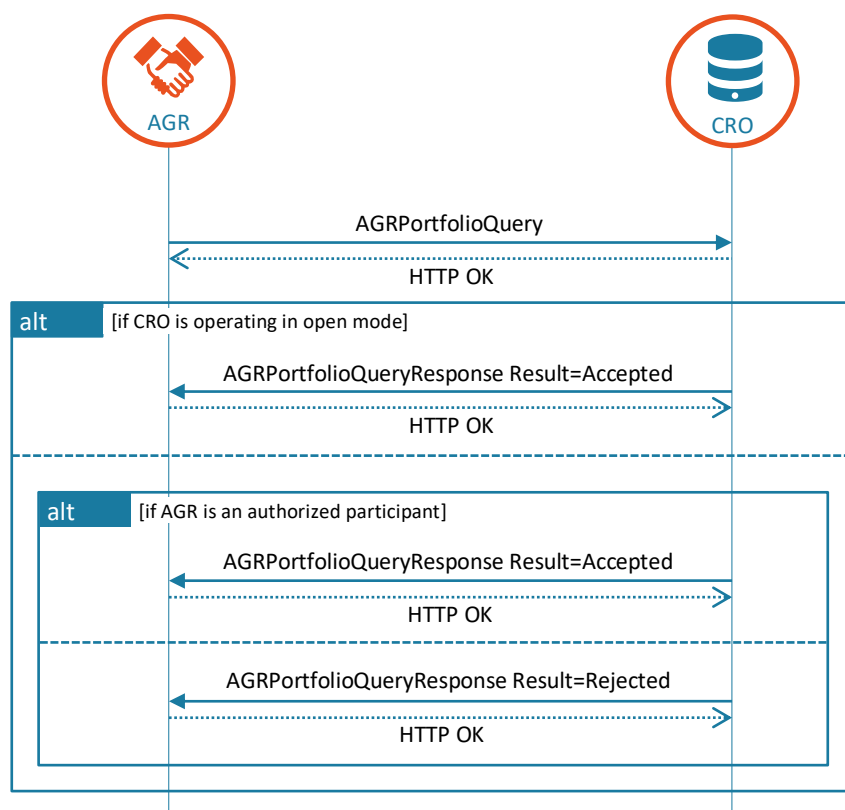


Figure 3-3: Retrieval of Congestion Points corresponding to AGR's connections

| Retrieve Congestion Points / AGR query | | |
|--|---|--|
| Goal in context | Retrieve a list of all registered Connections represented by this AGR, grouped by Congestion Point, in order to enable flex trading with the responsible DSO. | |
| Preconditions | The AGR has registered the Connection identifiers for which it represents Prosumers in the Common Reference. | |
| Successful outcome | The AGR receives a list of Connections, grouped by Connection Point and responsible DSO, and a list of Connections that have not been allocated to a DSO. | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | AGRPortfolioQuery failed to pass validation by the CRO |
| | Unauthorized | CRO is operating in closed mode and the AGR is not pre-registered as an authorized participant |
| | No connections available | The AGR has no registered connections at the Common Reference |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

AGRs will only obtain DSO identities, congestion point identifiers and connection identifiers for those connections they have registered in the common reference themselves.

Registered connections that have not (yet) been allocated by a DSO are returned in a separate list, to inform the AGR that there is no DSO available to trade with at those connections.

3.7.2 Retrieve Active Aggregators

If operating in open mode, the CRO will accept queries from any USEF-compliant participants implementing the AGR role. In closed mode, participants will need to be pre-configured in order for updates to be accepted.

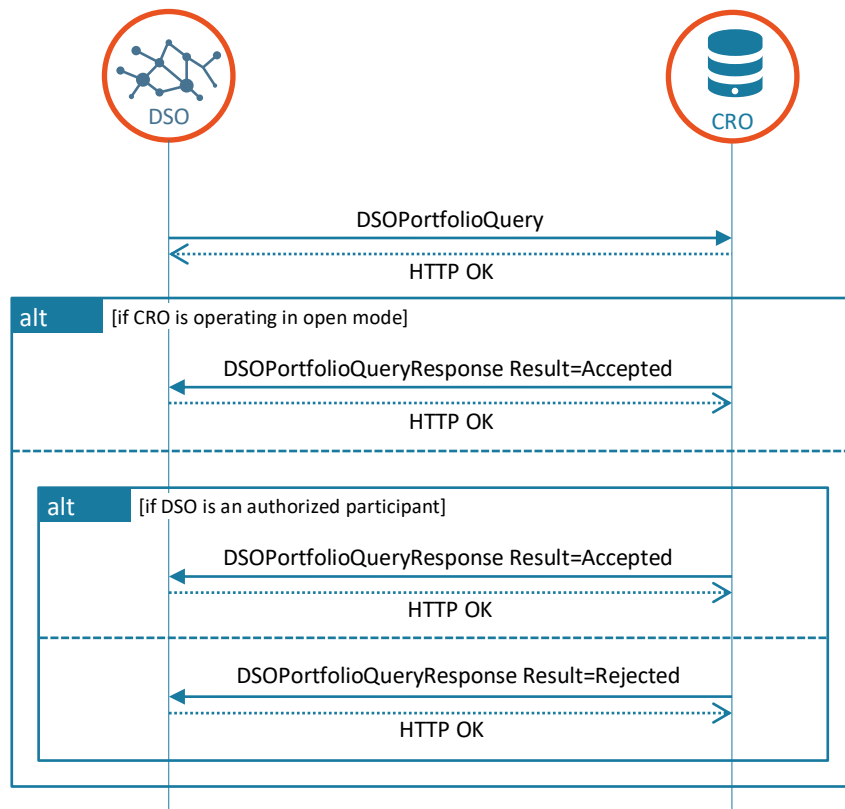


Figure 3-4: Retrieval of registered Connections, grouped by Congestion Point, including corresponding AGR identity

| Retrieve active Aggregators / DSO query | | |
|---|---|--|
| Goal in context | <p>Retrieve a list of all DSO-registered Congestion Points with, for each such point, a list of AGRs representing Prosumers there, including the number of Connections each AGR represents.</p> <p>The DSO then knows from which AGRs D-prognoses can be expected, and which percentage of the total Connections on each Congestion Point is affected by those prognoses.</p> | |
| Preconditions | <p>The DSO has determined its Congestion Points and published this information, including the associated Connection identifiers in the Common Reference.</p> <p>The AGR has registered the Connection identifiers for which it represents Prosumers in the Common Reference (see UC1027)</p> | |
| Successful outcome | AGRs which represent Prosumers at any of the Congestion Points registered by the DSO are available to the DSO | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | DSOPortfolioQuery failed to pass validation by the CRO |
| | Unauthorized | CRO is operating in closed mode and the DSO is not pre-registered as an authorized participant |
| | No connections available | The DSO has no registered connections at the Common Reference |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

DSOs may only obtain AGR identities and combined connection counts for those congestion points they have registered in the common reference themselves.

3.7.3 Exchange Flexibility Reservation Update

Where bilateral contracts are used, the FlexReservationUpdate message can be used at this stage to set or release reserved flexibility.

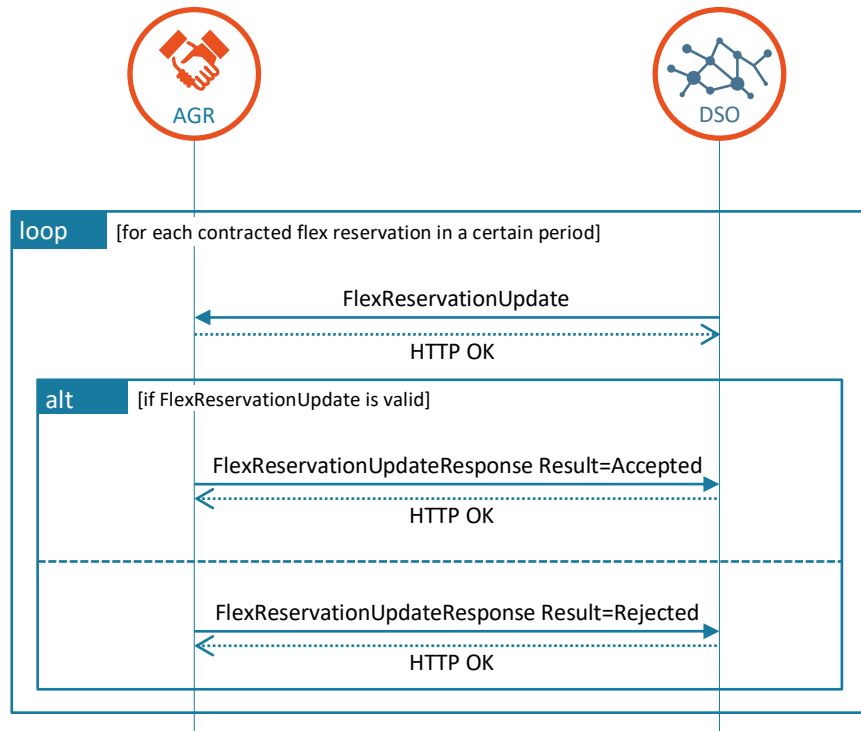


Figure 3-5: Exchange of FlexReservationUpdate

| FlexReservationUpdate | | |
|---------------------------|--|---|
| Goal in context | For all reserved power in a bilateral contract, the DSO signals which part of the contracted volume is still reserved and which part is not needed and may be used for other purposes. | |
| Preconditions | The DSO and AGR have agreed on a bilateral contract (out of scope for UFTP) which the FlexReservationUpdate refers to. | |
| Successful outcome | The AGR has received the update and can potentially reoptimize its portfolio based on the changes. The DSO registers the update for settlement purposes. | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexReservationUpdate failed to pass validation by the AGR |
| | Deadline expired | The deadline for the DSO to release reserved flexibility has expired |
| | Power value rejection | One or more Power values in the FlexReservationUpdate are not conform agreement |
| | [User defined] | Any other reasonable cause to reject the message |

3.8 Use cases – Validate phase

In Section 2.4, informative description of the validate phase processes has been given. In this chapter the use cases will be described as derived from the validate phase.

The USEF MCM validate phase specifies the following use cases:

Table 3-4: Use cases for the Validate phase.

| Name | Direction | Message types |
|---|-----------|---|
| Exchange D-Prognoses per Congestion Point | AGR → DSO | Prognosis / PrognosisResponse |
| Exchange Flexibility Requests | AGR ← DSO | FlexRequest / FlexRequestResponse |
| Exchange Flexibility Offers | AGR → DSO | FlexOffer / FlexOfferResponse |
| Revocation Flexibility Offer | AGR → DSO | FlexOfferRevocation / FlexOfferRevocationResponse |
| Exchange Flexibility Orders | AGR ← DSO | FlexOrder / FlexOrderResponse |

The use cases are explained in the following sections.

3.8.1 Exchange D-Prognoses per Congestion Point

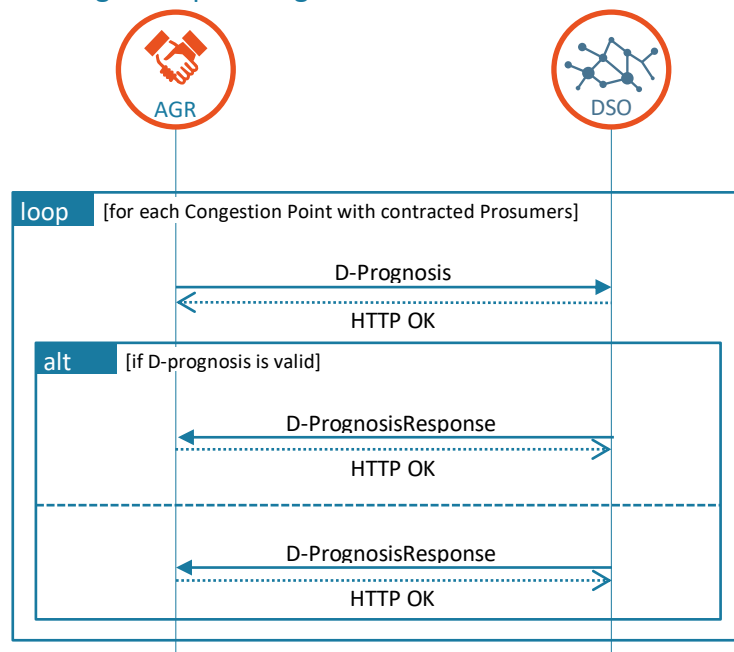


Figure 3-6: Exchange of D-prognoses

| D-prognosis | | |
|--------------------|--|--|
| Goal in context | D-prognoses are (re)created by the AGR per congestion point, based on its portfolio. These D-prognoses will be submitted to the DSO. D-prognoses should be submitted at least once before the gate closure time. D-prognoses can be updated by transmitting a new one. | |
| Preconditions | Congestion point(s) defined and retrieved from the CRO, AGR per congestion point retrieved from CRO by DSO, AGR-DSO contracts in place. | |
| Successful outcome | D-prognosis message created and accepted by the DSO for each congestion point on which the AGR has active Connections | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | D-prognosis failed to pass validation by the DSO |
| | Lacking ISPs | The prognosis does not include all ISPs in the Period it applies to |
| | Power value rejection | One or more Power values in the prognosis are not plausible |
| | Subordinate sequence number | The message sequence is lower than that of a previously received D-prognosis |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

Congestion points (with associated DSOs) at which the AGR has contracted prosumers, and active AGRs, are available via the common reference and should have been retrieved prior to executing this use case.

It is essential to use a sequence number that is incremented each time a new revision is sent so the order of transmission can be traced.

3.8.2 Exchange Flexibility Requests

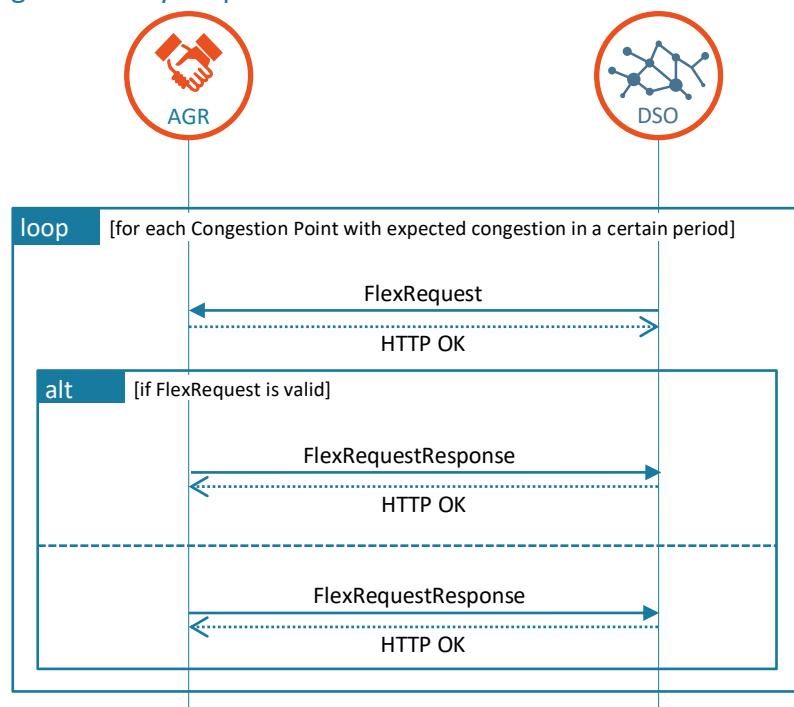


Figure 3-7: Exchange of FlexRequests

| FlexRequest | | |
|---------------------------|---|--|
| Goal in context | If grid congestion is expected, requests for flexibility are created by the DSO and sent to AGRs potentially capable of helping to reduce that congestion. These requests serve as the basis for subsequent flexibility offers. | |
| Preconditions | AGR-DSO market contract in place. One or more Congestion Points registered in the Common Reference, with at least one AGR with contracted Connections on that point. DSO grid safety analysis performed. | |
| Successful outcome | Flexibility requests submitted to AGR(s) | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexRequest failed to pass validation by the AGR |
| | Lacking Requested Disposition | The FlexRequest has no ISP with a “requested” disposition. |
| | Requested Power discrepancy | One or more ISPs with a “requested” disposition has no direction: MinPower < 0 and MaxPower > 0. |
| | Power discrepancy | One or more ISPs has a higher MinPower than MaxPower value. |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

Each flexibility request should include the ISPs for which congestion is expected, including a direction ($\text{MinPower} \geq 0$ or $\text{MaxPower} \leq 0$), and it is advisable to send multiple ISPs for which capacity is still available, to allow AGRs to time-shift load. For each period during which congestion is expected, the DSO may create any number of flexibility requests (including no requests at all). If multiple requests are created, their content must not be identical. Which variations (more/less requested power per ISP, time-shift of load) are created is a DSO-internal business decision.

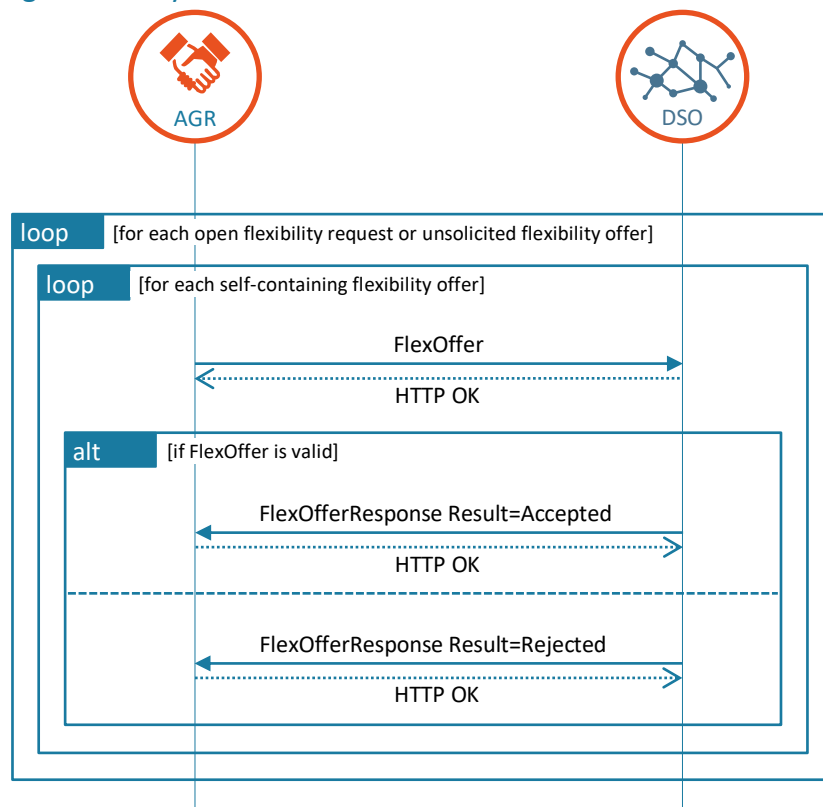
3.8.3 Exchange Flexibility Offers

Figure 3-8: Exchange of FlexOffers

| FlexOffer | | |
|---------------------------|---|---|
| Goal in context | <p>The AGR creates flexibility offers based on previous DSO flexibility requests and sends these to the originating DSOs. It is also possible to send FlexOffers that do not originate from a FlexRequest, which are called unsolicited FlexOffers.</p> <p>Each offer indicates to which extent the AGR can provide flexibility, what the effect on its baseline will be and what the price for the flexibility would be.</p> | |
| Preconditions | <p>AGR-DSO market contract in place.</p> <p>AGR has at least one contracted Connection on the Congestion point.</p> <p>Flexibility offers can be based on previous flexibility requests that have been accepted and stored, or be initiated without a preliminary flex request (called an unsolicited flex offer).</p> | |
| Successful outcome | Flexibility offers sent to the DSO | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexOffer failed to pass validation by the DSO |
| | No baseline | The AGR has no valid baseline, possibly lacking a valid D-prognosis |
| | Power value rejection | One or more Power values in the FlexOffer are not plausible |

| | | |
|--|------------------------|--|
| | Request mismatch | None of the ISPs with a “requested” disposition in the referred FlexRequest, is mentioned in the FlexOffer |
| | No MutEx offer support | The FlexOffer contains mutual exclusive offers (OfferOptions), which is not supported by the DSO |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

- USEF does not specify how the AGR determines the optimal flexibility offer, offer price or optimal timing to send the offer to the DSO.
- AGRs may choose to send multiple FlexOffer messages: this includes sending a FlexOffer that better meets the DSO request at a much later time when circumstances have changed. Unsolicited FlexOffers are also allowed, in which case the FlexRequestOrigin and FlexRequestSequence fields are empty.
- Sending a FlexOffer for partial activation is always allowed, although it is ignored by the DSO if it does not support partial activation (in which case the MinActivationFactor is always 1).

Note that acceptance of the FlexOffer message does not imply ordering of the flexibility.

3.8.4 Revocation Flexibility Offer

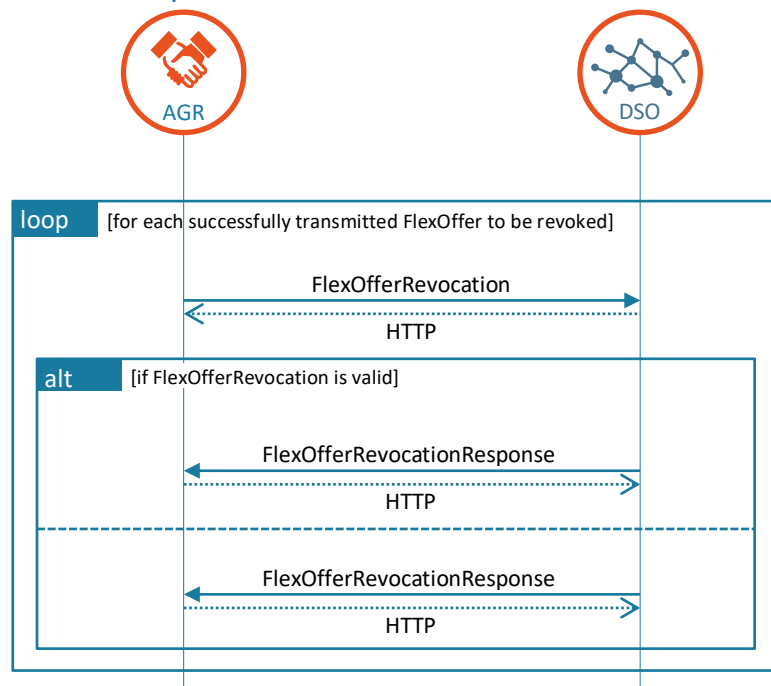


Figure 3-9: Revocation of FlexOffer

| Revoke FlexOffer | | |
|---------------------------|--|--|
| Goal in context | When a previously sent flexibility offer is no longer valid, despite the validity period of the offer not having expired yet, the AGR informs the DSO that the offer is revoked. After revocation, the DSO cannot order the flexibility from this offer anymore. | |
| Preconditions | The concerning flexibility offer has been sent to and acknowledged by the DSO, and has not (yet) been procured using a FlexOrder | |
| Successful outcome | Flexibility offer revocation notification submitted to the DSO | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexOfferRevocation failed to pass validation by the DSO |

| | | |
|--|----------------------|--|
| | Flexibility procured | The FlexOffer cannot be revoked because its flexibility has already been ordered |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

Only flexibility offers communicated using a FlexOffer message which has already been acknowledged by the DSO can be revoked. If the acknowledgement is still pending, the AGR should delay sending its FlexOfferRevocation message until the acknowledgement has been received. When the DSO receives a revocation of a flexibility offer it wants to order, this is not a valid base to reject the revocation. After revocation, the offer can no longer be ordered.

FlexOffers may be revoked at any time in the plan and validate phases (and also in operate phase as described in Section 3.9.2), unless there is already a FlexOrder based on the offer. Notice that FlexOrders and FlexOfferRevocations can cross each other. Where this is the case, priority should be given to the FlexOfferRevocation.

3.8.5 Exchange Flexibility Orders

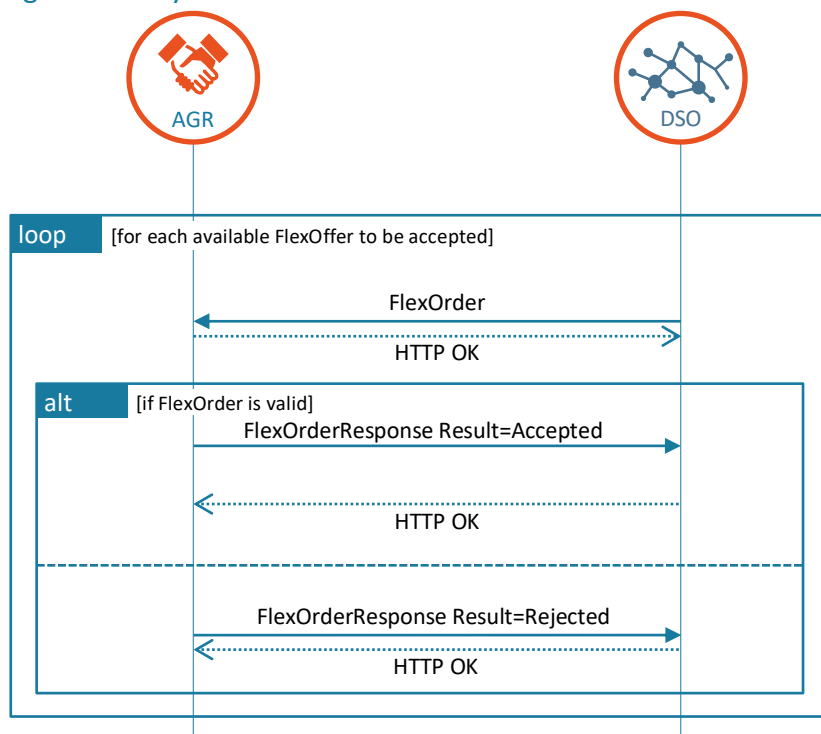


Figure 3-10: Exchange of FlexOrder

| FlexOrder | | |
|--------------------|---|--|
| Goal in context | Process for the DSO to accept (some) flexibility offers to solve Congestion based on the Prognosis. | |
| Preconditions | Corresponding FlexOffer has been received and accepted by DSO and is still valid. | |
| Successful outcome | Flexibility procured | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexOrder failed to pass validation by the AGR |
| | ISP mismatch | ISPs from the FlexOrder do not match the ISPs given in the FlexOffer |
| | Power mismatch | Ordered flexibility does not match the offered flexibility |
| | Price mismatch | Price in the order does not match the price given in the offer |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

The DSO needs to make its own decisions about which FlexOffers to accept. FlexOffers can only be ordered once and have to be ordered as specified in the corresponding offer for the specified price. Only in situations where partial activation is supported and offered in the FlexOffer can the ISP power and price be altered to conform to the specifications from Section 2.4.4. After a successful FlexOrder transmission, the AGR is responsible for providing the ordered flexibility via an updated D-prognosis which reflects the traded flexibility.

3.9 Use cases - Operate phase

In Section 2.5 the informative description of the operate phase processes are provided. In this chapter, the use cases will be described as derived from the operate phase.

The USEF MCM operate phase specifies the following use cases:

Table 3-5: Use cases for the operate phase.

| Name | Direction | Message types |
|-------------------------------------|-----------|---|
| Exchange updated D-Prognoses | AGR → DSO | Prognosis / PrognosisResponse |
| Revocation Flexibility Offer | AGR → DSO | FlexOfferRevocation / FlexOfferRevocationResponse |
| Exchange Flexibility Orders | AGR ← DSO | FlexOrder / FlexOrderResponse |

The use cases are explained in the following sections.

3.9.1 Exchange updated D-Prognoses

The process diagram is similar to Figure 3-6.

| D-prognosis | | |
|---------------------------|---|--|
| Goal in context | Changes in D-prognoses to be realized are transmitted. | |
| Preconditions | AGR has created initial D-prognoses in the Plan/Validate phases, and has subsequently detected deviations in Operate that resulted in a portfolio state that requires those prognoses to be re-created. | |
| Successful outcome | Prognosis sent to DSO that reflects all relevant changes | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | D-prognosis failed to pass validation by the DSO |
| | Lacking ISPs | The prognosis does not include all ISPs in the Period it applies to |
| | Power value rejection | One or more Power values in the prognosis are not plausible |
| | Subordinate sequence number | The message sequence is lower than that of a previously received D-prognosis |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

The process is similar to that described in Section 3.8.1.

Deviations can be caused by new flexibility orders from the DSO and by unforeseen change of behavior by assets.

The basic process for re-creating D-prognoses is the same as that used to create them in the first place, in the plan/validate phases.

When re-creating prognoses in the operate phase, values for ISPs that are already in the past should remain unchanged, as these will be ignored by DSOs anyway.

3.9.2 Revocation Flexibility Offer

The process diagram is similar to Figure 3-9.

| Revoke FlexOffer | | |
|---------------------------|--|--|
| Goal in context | Inform the DSO that a previously sent flexibility offer is no longer valid, despite the validity period of the offer not having expired yet. | |
| Preconditions | A flexibility offer has been sent to and acknowledged by the DSO | |
| Successful outcome | Flexibility offer revocation notification submitted to the DSO | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexOfferRevocation failed to pass validation by the DSO |
| | Flexibility procured | The FlexOffer cannot be revoked because its flexibility has already been ordered |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

The process is similar to that described in Section 3.8.4.

Flexibility offers may be revoked in the operate phase provided that none of the ISPs in the period the offer applies to are already in, or past, the operate phase.

3.9.3 Exchange Flexibility Orders

This process is triggered when flexibility is required to resolve congestion detected during the operate phase. This might, for example, be the case when the prognoses change after the validate phase and there are still FlexOffers available.

The process diagram is similar to Figure 3-10.

| FlexOrder | | |
|---------------------------|--|--|
| Goal in context | Accept (some) flexibility offers to solve congestion during Operate phase. | |
| Preconditions | Congestion detected while monitoring the grid, corresponding FlexOffer has been received and accepted by DSO and is still valid. | |
| Successful outcome | Flexibility procured | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexOrder failed to pass validation by the AGR |
| | ISP mismatch | ISPs from the FlexOrder do not match the ISPs given in the FlexOffer |
| | Power mismatch | Ordered flexibility does not match the offered flexibility |
| | Price mismatch | Price in the order does not match the price given in the offer |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

The process is similar to that described in Section 3.8.5.

Note that where a FlexOrder is rejected, manual processing is unlikely to result in timely resolution at this stage.

Note that although the updated D-prognosis is unlikely to be of immediate use to the DSO, it's definitely required for settlement.

3.10 Use cases - Settle phase

The informative description of the settle phase processes are provided in Section 2.6. In this chapter, the use cases will be described, as derived from the settle phase.

The USEF MCM settle phase specifies the following use case:

Table 3-6: Use cases for the settle phase.

| Name | Direction | Message types |
|--------------------------|-----------|--|
| Process Settlement Items | AGR ← DSO | FlexSettlement/ FlexSettlementResponse |

3.10.1 Process Settlement Items

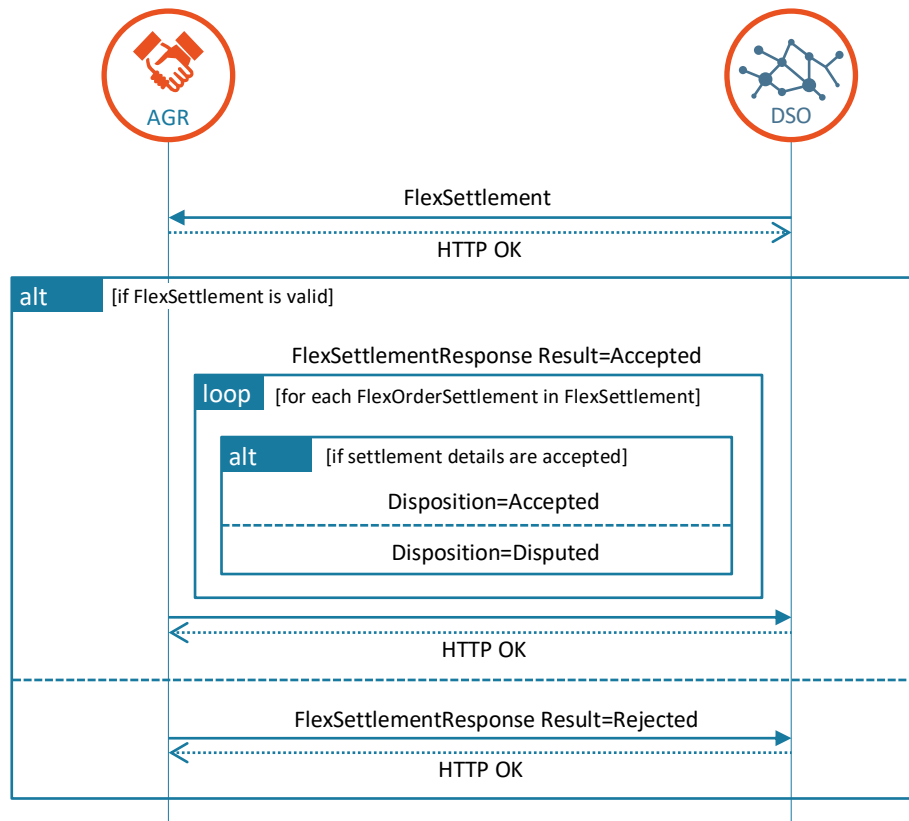


Figure 3-11: Transmission of settlement including acceptance process per settlement item

| Settlement | | |
|--------------------|---|---|
| Goal in context | DSO sends consolidated settlement volumes and prices to the AGR for a specific period. AGR verifies the settlement calculations performed by DSO: acknowledge settlement items that are plausible according to the AGR or start a dispute process if they show disputable results. AGR returns the list of verified settlement items. | |
| Preconditions | All settlement items within the period are available and complete | |
| Successful outcome | Periodic settlement invoice sent by DSO, validated by AGR and a potential dispute process is started. | |
| Failure outcome | RejectionReason | Cause of rejection |
| | <See section 3.5> | FlexSettlement failed to pass validation by the AGR |
| | Missing Settlement Items | Not all flexibility that have been procured within the given period is included as a settlement item. |
| | PeriodStart rejected | The settlement period starts too late (for example in case there are unsettled items from an earlier period) or is not plausible (for example in case it starts in the future). |

| | | |
|--|--------------------|---|
| | PeriodEnd rejected | The settlement period ends too early (for example in case there are unsettled items from a later period that have to be treated too) or is not plausible (for example in case it is earlier than the PeriodStart or in the future). |
| | [User defined] | Any other reasonable cause to reject the message |

Related information

If settlement items are missing, the message should be rejected (as mentioned in failure outcome). This is to encourage sending of all required information necessary to settle trades as soon as possible, thereby avoiding a series of unsettled trades for a given period.

Note that the rejection of a FlexSettlement message is not a means of raising a dispute for settlement items. The dispute process is out-of-scope for USEF.

4 Message descriptions

4.1 Scope

USEF defines the content of XML messages between role processes, as required for implementation of USEF's MCM. Each message has a unique recipient, identified by the Internet domain and USEF role of the participant. The message transfer mechanism is responsible for sending and receiving these messages.

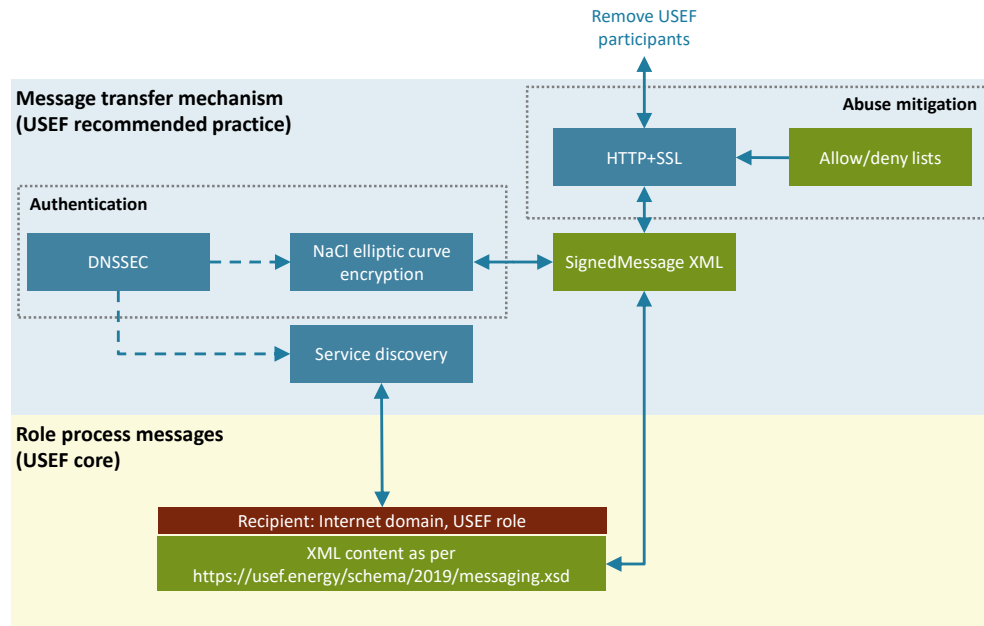


Figure 4-1: Relationship between the normative USEF XML messages (yellow pane) and the best-practice message transfer mechanism (light-blue pane).

When selecting a message transfer mechanism, local market conditions should be considered and the USEF privacy and security guidelines should be followed [5]. Further recommendations about the message transport mechanism are given in Appendix 3.

4.2 Message catalog

USEF messages consist of XML, use UTF-8 encoding, and should validate against the USEF schema corresponding to the specification version implemented by a participant. The current version of this schema, as well as historic production versions are available for download from the [Shapeshifter Github](#). Message types can be differentiated using the name of their root node.

To allow extensibility as well as forward and backward compatibility of XML messages, the USEF schema allows additional elements in each defined sequence, as long as these elements have an explicit, non-default XML namespace.

4.2.1 Power

Power values occur in many USEF messages. The sign of a power value is used to distinguish between production and consumption, from the perspective of the prosumer. A positive value indicates that power flows towards the prosumer (consumption), a negative value indicates flow towards the grid (production).

The following statements are true for ISP elements in all USEF messages:

- Each ISP in a period may only be addressed once.
- ISP elements may not cross period boundaries.

4.2.2 Metadata attributes

The Metadata attributes contain metadata that are common for each non-wrapper USEF message.

| XML representation summary | | |
|------------------------------|---|-----------------------------|
| <code>SenderDomain</code> | = | <code>InternetDomain</code> |
| <code>RecipientDomain</code> | = | <code>InternetDomain</code> |
| <code>TimeStamp</code> | = | <code>DateTime</code> |
| <code>MessageID</code> | = | <code>UUID</code> |
| <code>ConversationID</code> | = | <code>UUID</code> |
| <code>ISP-Duration</code> | = | <code>Duration</code> |
| <code>TimeZone</code> | = | <code>TimezoneName</code> |
| <code>Version</code> | = | <code>SpecVersion</code> |

| | |
|-----------------|--|
| SenderDomain | The Internet domain of the USEF participant sending this message. When receiving a message, its value should match the value specified in the SignedMessage wrapper: otherwise, the message must be rejected as invalid. When replying to this message, this attribute, combined with the SenderRole, is used to look up the USEF endpoint the reply message should be delivered to. |
| RecipientDomain | Internet domain of the participant this message is intended for. When sending a message, this attribute, combined with the RecipientRole, is used to look up the USEF endpoint the message should be delivered to. |
| TimeStamp | Date and time this message was created, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime). |
| MessageID | Unique identifier (UUID/GUID as per IETF RFC 4122) for this message, to be generated when composing each message. |
| ConversationID | Unique identifier (UUID/GUID as per IETF RFC 4122) used to correlate responses with requests, to be generated when composing the first message in a conversation and subsequently copied from the original message to each reply message. |
| ISP-Duration | ISO 8601 time interval (minutes only, for example PT15M) indicating the duration of the ISPs referenced in this message. Although the ISP length is a market-wide fixed value, making this assumption explicit in relevant messages is important for validation purposes, allowing implementations to reject messages with an errant ISP duration. Only messages that refer to individual ISPs must include this attribute in the metadata. |
| TimeZone | Time zone ID (as per the IANA time zone database, http://www.iana.org/time-zones , for example: Europe/Amsterdam) indicating the UTC offset that applies to the Period referenced in this message. Although the time zone is a market-wide fixed value, making this assumption explicit in relevant messages is important for validation purposes, allowing implementations to reject messages with an errant UTC offset. Only messages that refer to a specific Period must include this attribute in the metadata. |
| Version | Shapeshifter specification version following semantic versioning rules e.g. 3.0.0 |

4.2.3 SignedMessage

The SignedMessage element represents the secure wrapper used to submit USEF XML messages from the local message queue to the message queue of a remote participant. It contains minimal metadata (which is distinct from the common metadata used for all other messages), allowing the recipient to look up the sender's cryptographic scheme and public keys, and the actual XML message, as transformed (signed/sealed) using that cryptographic scheme.

| XML representation summary | | |
|--------------------------------|---|--------------------------------------|
| <code><SignedMessage</code> | | |
| <code>SenderDomain</code> | = | <code>InternetDomain</code> |
| <code>SenderRole</code> | = | <code>("AGR" "CRO" "DSO")</code> |
| <code>Body</code> | = | <code>base64Binary</code> |
| <code>/></code> | | |

| | |
|--------------|---|
| SenderDomain | The Internet domain of the USEF participant sending this message. Upon receiving a message, the recipient should validate that its value matches the corresponding attribute value specified in the inner XML message, once un-sealed: if not, the message must be rejected as invalid. Domain name matching regular expression <code>^([a-z0-9]+(-[a-z0-9]+)*\.)+[a-z]{2,}\$</code> |
| SenderRole | The USEF role of the participant sending this message: AGR, CRO or DSO. Receive-time validation should take place as described for the SenderDomain attribute above. The USEF specification identifies additional roles such as BRP and MDC. These are however not in scope of UFTP. |
| Body | The Base-64 encoded inner XML message contained in this wrapper, as transformed (signed/sealed) using the sender's cryptographic scheme. The recipient can determine which scheme applies using a DNS or configuration file lookup, based on the combination of SenderDomain and SenderRole. |

4.2.4 TestMessage

TestMessage does not have any content (other than the mandatory metadata) and may be used by implementations for (connectivity) testing purposes.

XML representation summary

```
< TestMessage
  Metadata...
/>
```

| | |
|----------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
|----------|--|

4.2.5 TestMessageResponse

Upon receiving a TestMessage, the receiving implementation must reply with a TestMessageResponse. Like the TestMessage itself, the TestMessageResponse does not have any content (other than the mandatory metadata).

XML representation summary

```
< TestMessageResponse
  Metadata...
/>
```

| | |
|----------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
|----------|--|

4.2.6 AGRPortfolioUpdate

The AGRPortfolioUpdate is used by the AGR to indicate on which Connections it represents prosumers.

XML representation summary

```
<AGRPortfolioUpdate
  Metadata...
  <Connection (0..n)
    EntityAddress = EntityAddress
    StartPeriod = Date
    EndPeriod = Date (optional)
  />
/>
```

| | |
|---------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| Connection | |
| EntityAddress | EntityAddress of the Connection entity being updated. |
| StartPeriod | The first Period hat the AGR represents the prosumer at this Connection. |
| EndPeriod | The last Period that the AGR represents the prosumer at this Connection, if applicable. |

4.2.7 AGRPortfolioUpdateResponse

Upon receiving and processing an AGRPortfolioUpdate message, the receiving implementation must reply with an AGRPortfolioUpdateResponse, indicating whether the update was handled successfully.

XML representation summary

```
< AGRPortfolioUpdateResponse
  Metadata...
  AGRPortfolioUpdateMessageID = UUID
  Result                       = ("Accepted" | "Rejected")
  RejectionReason              = String (only if Result = "Rejected")
/>
```

| | |
|-----------------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| AGRPortfolioUpdateMessageID | MessageID of the AGRPortfolioUpdate message |
| Result | Indication whether the AGRPortfolioUpdate was accepted or rejected. |
| RejectionReason | In case the AGRPortfolioUpdate was rejected, this attribute must contain a human-readable description of the reason. |

4.2.8 AGRPortfolioQuery

The AGRPortfolioQuery is used by the AGR to retrieve additional information on the connections.

XML representation summary

```
< AGRPortfolioQuery
  Metadata...
  Period = Date
/>
```

| | |
|----------|--|
| Metadata | The metadata for this message. For details, see section 4.4.3. |
| Period | The Period for which the AGR requests the portfolio information. |

4.2.9 AGRPortfolioQueryResponse

Upon receiving and processing an AGRPortfolioQuery message, the receiving implementation must reply with an AGRPortfolioUpdate, indicating whether the query was handled successfully.

XML representation summary

```
<AGRPortfolioQueryResponse
  Metadata...
  AGRPortfolioQueryMessageID = UUID
  Result                     = ("Accepted" | "Rejected")
  RejectionReason            = String (only if Result = "Rejected")
  Period                     = Date
  <DSO-View                  (only if Result = "Accepted")
    <DSO-Portfolio            (0...n)
```

```

DSO-Domain          = InternetDomain
<CongestionPoint    (1...n)
  EntityAddress      = EntityAddress
  MutexOffersSupported = Boolean
  DayAheadRedispatchBy = ("AGR" | "DSO" )
  IntradayRedispatchBy = ("AGR" | "DSO" ) (optional)
  <Connection        = (1...n)
    EntityAddress      = EntityAddress
  />
/>
<Connection          = (0...n)
  EntityAddress        = EntityAddress
/>
/>
/>
/>
/>

```

| | |
|----------------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| AGRPortfolioQueryMessageID | MessageID of the AGRPortfolioQuery message |
| Result | Indication whether the AGRPortfolioQuery was accepted or rejected. |
| RejectionReason | In case the AGRPortfolioQuery was rejected, this attribute must contain a human-readable description of the reason. |
| Period | The Period that the portfolio is valid. |
| DSO-View | |
| DSO-Portfolio | Portfolio of a single DSO that shares connections with the AGR |
| DSO-Domain | The InternetDomain of the DSO the portfolio applied to. |
| CongestionPoint | A CongestionPoint that contains at least one Connection of a prosumer that is represented by the AGR. |
| EntityAddress | Entity Address of the CongestionPoint entity being updated. |
| MutexOffersSupported | Indicates whether the DSO accepts mutual exclusive FlexOffers on this CongestionPoint |
| DayAheadRedispatchBy | Indicates which party is responsible for day-ahead redispatch, AGR or DSO. |
| IntradayRedispatchBy | Indicates which party is responsible for intraday ahead redispatch, AGR or DSO. If not specified, there will be no intraday trading on this CongestionPoint. |
| Connection | |
| EntityAddress | Entity Address of a Connection that is part of this CongestionPoint. |
| Connection | Connection not belonging to any DSO CongestionPoint |
| EntityAddress | Entity Address of the Connection |

4.2.10 DSOPortfolioUpdate

The DSOPortfolioUpdate is used by the DSO to indicate on which congestion points it wants to engage in flexibility trading.

XML representation summary

```

<DSOPortfolioUpdate
  Metadata...
  <CongestionPoint      (0...n)
    EntityAddress        = EntityAddress
    StartPeriod          = Date

```



```

    EndPeriod          = Date (optional)
    MutexOffersSupported = Boolean
    DayAheadRedispatchBy = ( "AGR" | "DSO" )
    IntradayRedispatchBy = ( "AGR" | "DSO" ) (optional)
    <Connection
      EntityAddress      = EntityAddress
      StartPeriod        = Date
      EndPeriod          = Date (optional)
    />
  />
/>

```

| | |
|----------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| CongestionPoint | |
| EntityAddress | Entity Address of the CongestionPoint entity being updated. |
| StartPeriod | The first Period that the DSO is trading on this CongestionPoint. |
| EndPeriod | The last Period that the DSO is trading on this CongestionPoint. |
| MutexOffersSupported | Indicates whether the DSO accepts mutual exclusive FlexOffers on this CongestionPoint |
| DayAheadRedispatchBy | Indicates which party is responsible for day-ahead redispatch, AGR or DSO. |
| IntradayRedispatchBy | Indicates which party is responsible for intraday ahead redispatch, AGR or DSO. If not specified, there will be no intraday trading on this CongestionPoint. |
| Connection | |
| EntityAddress | Entity Address of a Connection that is part of this CongestionPoint. |
| StartPeriod | <p>The first Period that the Connection is part of this CongestionPoint.</p> <p>Notes:</p> <p>The StartPeriod of a Connection should be greater than or equal to the StartPeriod of the CongestionPoint.</p> <p>The StartPeriod of a Connection should be smaller than or equal to the EndPeriod of the CongestionPoint.</p> <p>CRO implementors may choose to explicitly check for this and reject updates that do not comply.</p> |
| EndPeriod | <p>The last Period that the Connection is part of this CongestionPoint .</p> <p>Notes:</p> <p>The EndPeriod of a Connection should be null if and only if the EndPeriod of the CongestionPoint is null.</p> <p>The EndPeriod of a Connection should be greater than or equal to the StartPeriod of the CongestionPoint.</p> <p>The EndPeriod of a Connection should be smaller than or equal to the EndPeriod of the CongestionPoint.</p> <p>CRO implementors may choose to explicitly check for this and reject updates that do not comply.</p> |

4.2.11 DSOPortfolioUpdateResponse

Upon receiving and processing a DSOPortfolioUpdate message, the receiving implementation must reply with a DSOPortfolioUpdateResponse, indicating whether the update was handled successfully.

XML representation summary

```

<DSOPortfolioUpdateResponse
  Metadata...
  DSOPortfolioUpdateMessageID = UUID
  Result                       = ("Accepted" | "Rejected")
  RejectionReason              = String (only if Result = "Rejected")
/>

```

| | |
|-----------------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| DSOPortfolioUpdateMessageID | MessageID of the DSOPortfolioUpdate message |
| Result | Indication whether the DSOPortfolioUpdate was accepted or rejected. |
| RejectionReason | In case the DSOPortfolioUpdate was rejected, this attribute must contain a human-readable description of the reason. |

4.2.12 DSOPortfolioQuery

DSOPortfolioQuery is used by DSOs to discover which AGRs represent connections on its registered congestion point(s).

XML representation summary

```

<DSOPortfolioQuery
  Metadata...
  EntityAddress                = EntityAddress
  Period                      = Date
/>

```

| | |
|---------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| EntityAddress | Entity Address of the CongestionPoint entity being queried. |
| Period | The Period being queried. |

4.2.13 DSOPortfolioQueryResponse

Upon receiving and processing a DSOPortfolioQuery message, the receiving implementation must reply with a DSOPortfolioQueryResponse, indicating whether the query executed successfully, and if it did, including the query results. Most queries will return zero or more congestion points

XML representation summary

```

<DSOPortfolioQueryResponse
  Metadata...
  DSOPortfolioQueryMessageID = UUID
  Result                     = ("Success" | "Failure")
  RejectionReason            = String (only if Result = "Failure")
  Period                    = Date
  <CongestionPoint
    EntityAddress            = EntityAddress
    <Connection
      EntityAddress          = EntityAddress
      AGR-Domain             = InternetDomain (optional)
    />
  />
/>

```

| | |
|----------------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| DSOPortfolioQueryMessageID | MessageID of the DSOPortfolioQuery message |
| Result | Indication whether the query was executed successfully or failed. |
| RejectionReason | In case the query failed, this attribute must contain a human-readable description of the failure reason. |
| Period | The period that was queried. This is also the Period for which the portfolio is valid. |
| CongestionPoint | The congestion point that was queried. |
| EntityAddress | The entity address of the congestion point. |
| Connection | A Connection that is part of the congestion point. |
| EntityAddress | The entity address of the Connection |
| AGR-Domain | The internet domain of the AGR that represents the prosumer connected on this Connection, if applicable. |

4.2.14 FlexReservationUpdate

For bilateral contracts, FlexReservationUpdate messages are used by DSOs to signal to an AGR which part of the contracted volume is still reserved and which part is not needed and may be used for other purposes. For each ISP, a power value is given which indicates how much power is still reserved. Zero power means that no power is reserved for that ISP and the sign of the power indicates the direction.

XML representation summary

```

<FlexReservationUpdate
  Metadata...
  Period                = Period
  CongestionPoint       = EntityAddress
  ContractID            = String
  ServiceType           = String (optional)
  Reference             = String
  Unit                  = PowerUnitType (optional)
  <ISP                  (1..n)
    Disposition         = "Available" | "Requested" (optional)
    Power               = Integer
    Start               = Integer
    Duration            = Integer (optional, default = 1)
  />
/>

```

| | |
|-----------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| Period | The Period this FlexRequest message applies to. |
| CongestionPoint | Entity Address of the Congestion Point this FlexRequest message applies to. |
| ContractID | Reference to the bilateral contract in question. |
| ServiceType | Service type for this request, the service type determines response characteristics such as latency or asset participation type. Each contract may specify multiple service-types. |
| Reference | Message reference, assigned by the DSO originating the FlexReservationUpdate. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| ISP | |
| Disposition | Indication whether the Power specified for this ISP represents available capacity or a request for reduction/increase. |

| | |
|----------|--|
| Power | Remaining reserved power specified for this ISP (in the specified `Unit`). See important notes about the flex reservation mechanism in Appendix 3. |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of the ISPs this element represents. Optional, default value is 1. |

4.2.15 FlexReservationUpdateResponse

Upon receiving and processing a FlexReservationUpdate message, the AGR must reply with a FlexReservationUpdateResponse, indicating whether the FlexReservationUpdate was processed successfully.

XML representation summary

```
<FlexReservationUpdateResponse
  Metadata...
  FlexReservationUpdateMessageID = UUID
  Result = ("Accepted" | "Rejected")
  RejectionReason = String (only if Result = "Rejected")
/>
```

| | |
|--------------------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexReservationUpdateMessageID | MessageID of the concerning FlexReservationUpdate message |
| Result | Indication whether the FlexReservationUpdate was accepted or rejected. Rejection is allowed in case the FlexReservationUpdate does not match the bilateral contract or its agreements or the deadline for sending the update has expired. |
| RejectionReason | In case the request was rejected, this attribute must contain a human-readable description of the reason. |

4.2.16 D-Prognosis

D-Prognosis messages are used to communicate D-prognoses between AGRs and DSOs. D-Prognosis messages always contain data for all ISPs for the period they apply to, even if a prognosis is sent after the start of the period, when one or more ISPs are already in the operate or settlement phase. Receiving implementations should ignore the information supplied for those ISPs.

XML representation summary

```
<D-Prognosis
  Metadata...
  Period = Period
  CongestionPoint = EntityAddress
  Revision = Long
  Unit = PowerUnitType (optional)
  <ISP (1..n)
    Power = Integer
    Start = Integer
    Duration = Integer (optional, default = 1)
  />
/>
```

| | |
|-----------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| Period | The Period this D-Prognosis applies to. |
| CongestionPoint | Entity Address of the Congestion Point this prognosis applies to. Required for D-prognoses, prohibited for other prognosis types. |

| | |
|----------|---|
| Revision | Revision of this message. A sequence number that must be incremented each time a new revision of a prognosis is sent. The combination of SenderDomain and PrognosisSequence should be unique. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| ISP | |
| Power | Power specified for this ISP (in the specified `Unit`). Also see the important notes about the sign of this attribute in the main documentation entry for the Power attribute. |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of the ISPs this element represents. |

Note: Every day-ahead D-Prognosis must contain a Power value for every single ISP of the specified Period.

4.2.17 D-PrognosisResponse

Upon receiving and processing a prognosis message, the receiving implementation must reply with a D-PrognosisResponse, indicating whether the prognosis was syntactically and semantically valid, and, where any FlexOrder messages have previously been sent for any ISP in the period the prognosis applies to, a FlexOrderStatus indication for each of those orders.

XML representation summary

```
<D-PrognosisResponse
  Metadata...
  D-PrognosisMessageID    =   UUID
  Result                  =   ("Accepted" | "Rejected")
  RejectionReason         =   String (Only if Result = "Rejected")
  <FlexOrderStatus        (0..n)
    Sequence              =   Long
    IsValidated           =   Boolean
  />
>
```

| | |
|----------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| D-PrognosisMessageID | Reference to the the D-prognosis that was accepted or rejected |
| Result | Indication whether the prognosis was accepted or rejected. A receiving implementation may reject a prognosis if it's syntactically invalid, uses a ISP duration or time zone that deviates from what is agreed upon for the market, is for a unknown Congestion Point or from an AGR that is not listed in the Common Reference, does not contain a full list of ISPs for the Period it applies to, contains implausible power values, applies to a Period for which all ISPs are already pending settlement, or if a Prognosis of the same type and from the same originator with a higher sequence number was already processed previously. |
| RejectionReason | In case the update was rejected, this attribute must contain a human-readable description of the failure. |
| FlexOrderStatus | Status of a previously submitted FlexOrder |
| Sequence | Sequence of the previously submitted FlexOrder. |
| IsValidated | Indication whether the Prognosis that was just received reflects this previously submitted FlexOrder. |

4.2.18 FlexRequest

FlexRequest messages are used by DSOs to request flexibility from AGRs. In addition to one or more ISP elements with Disposition=Requested, indicating the actual need to reduce consumption or production, the message should also include the remaining ISPs for the current period where Disposition=Available.

XML representation summary

```

<FlexRequest
  Metadata...
  Period                = Period
  ContractID            = String (optional)
  ServiceType           = String (optional)
  CongestionPoint       = EntityAddress
  Revision              = Long
  ExpirationDateTime    = DateTime
  Unit                  = PowerUnitType (optional)
  <ISP
    Disposition         = "Available" | "Requested"
    MinPower            = Integer
    MaxPower            = Integer
    Start               = Integer
    Duration            = Integer (optional, default = 1)
  />
/>

```

| | |
|--------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| Period | The Period this FlexRequest message applies to. |
| ContractID | Reference to the concerning contract, if applicable. The contract may be either bilateral or commoditized market contract. |
| ServiceType | Service type for this request, the service type determines response characteristics such as latency or asset participation type. Each contract may specify multiple service-types. |
| CongestionPoint | Entity Address of the Congestion Point this FlexRequest message applies to. |
| Revision | Revision of this message, a sequence number that must be incremented each time a new revision of a FlexRequest message is sent. |
| ExpirationDateTime | Date and time, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime) until when the FlexRequest message is valid. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| ISP | |
| Disposition | indication whether the Power specified for this ISP represents available capacity or a request for reduction/increase. At least one ISP should have Disposition = Requested |
| MinPower | Lower bound for available/requested space to deviate from the baseline (in the specified `Unit`). For further explanation, see section 2.4.4. Also see the important notes about the sign of this attribute in the ISP element, section 4.2.1. |
| MaxPower | Upper bound for available/requested space to deviate from the baseline (in the specified `Unit`). For further explanation, see section 2.4.4. Also see the important notes about the sign of this attribute in the ISP element, section 4.2.1. |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of the ISPs this element represents. Optional, default value is 1. |

4.2.19 FlexRequestResponse

Upon receiving and processing a FlexRequest message, the receiving implementation must reply with a FlexRequestResponse, indicating whether the flexibility request was processed successfully.

XML representation summary

```

<FlexRequestResponse
  Metadata...
  FlexRequestMessageID      = UUID
  Result                    = ("Accepted" | "Rejected")
  RejectionReason           = String (Only if Result = "Rejected")
/>

```

| | |
|----------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexRequestMessageID | MessageID of the FlexRequest that has been accepted or rejected. |
| Result | Indication whether the flex request was accepted or rejected. Rejection is allowed in case the FlexRequest is not based on our latest Prognosis, a FlexRequest from the same participant for the indicated period with a higher sequence number was already accepted previously or the FlexRequest does not contain any ISPs with Disposition=Requested. |
| RejectionReason | In case the request was rejected, this attribute must contain a human-readable description of the reason. |

4.2.20 FlexOffer

FlexOffer messages are used by AGRs to make DSOs an offer for provision of flexibility. A FlexOffer message contains a list of ISPs and, for each ISP, the change in consumption or production offered and the price for the total amount of flexibility offered. FlexOffer messages can be sent once a FlexRequest message has been received but can also be sent unsolicited. Note that multiple FlexOffer messages may be sent based on a single FlexRequest, e.g. to increase the chance that the DSO will order at least part of its available flexibility. The AGR must make sure that it can actually provide the flexibility offered across all of its FlexOffers.

XML representation summary

```

<FlexOffer
  Metadata...
  Period                = Period
  CongestionPoint       = EntityAddress
  ExpirationDateTime    = DateTime
  FlexRequestMessageID  = UUID (mandatory if and only if solicited)
  ContractID            = String (only if this offer refers to a bilateral
                           contract)
  ServiceType           = String (optional)
  D-PrognosisMessageID  = UUID (mandatory if and only if unsolicited)
  BaselineReference     = String (only if another baseline methodology is used)
  Currency              = ISO4217Currency
  Unit                  = PowerUnitType (optional)
  <OfferOption          (1..n)
    OptionReference     = String (only if there are multiple OfferOptions)
    Price               = ISO4217Currency
    MinActivationFactor = Number (optional [0.01-1.00])
    <ISP                (1..n)
      Power             = Integer
      Start              = Integer
      Duration          = Integer (optional, default = 1)
    />
  />
/>

```

| | |
|----------------------|--|
| Metadata | The metadata for this message. For details, see section 4.4.3. |
| Period | The Period the ISPs referenced in this FlexOffer message belong to. |
| CongestionPoint | Entity Address of the Congestion Point this FlexOffer message applies to. |
| ExpirationDateTime | Date and time, including the time zone (ISO 8601 formatted as per http://www.w3.org/TR/NOTE-datetime) until when the FlexOffer message is valid. |
| FlexRequestMessageID | MessageID of the FlexRequest message this request is based on. |
| ContractID | Reference to the concerning bilateral contract, if applicable. |
| ServiceType | Service type for this request, the service type determines response characteristics such as latency or asset participation type. Each contract may specify multiple service-types. |
| D-PrognosisMessageID | MessageID of the D-Prognosis this request is based on, if it has been agreed that the baseline is based on D-prognoses. |
| BaselineReference | Identification of the baseline prognosis, if another baseline methodology is used than based on D-prognoses. |
| Currency | ISO 4217 code indicating the currency that applies to the price of the FlexOffer. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| OfferOption | If the DSO does not support mutually exclusive offers it will reject FlexOffers that contain more than one OfferOption. |
| OptionReference | The identification of this option |
| Price | The asking price for the flexibility offered in this option. |
| MinActivationFactor | The minimal activation factor for this OfferOption. If this attribute is omitted, a default value of 1.00 must be assumed. An AGR may choose to include MinActivationFactor in FlexOffers even if the DSO is not interested in partial activation. In that case the DSO will simply use an ActivationFactor of 1.00 in every FlexOrder. |
| ISP | |
| Disposition | Optional, used only for FlexRequest messages: indication whether the Power specified for this ISP represents available capacity or a request for reduction/increase. |
| Power | Power specified for this ISP (in the specified `Unit`). Also see the important notes about the sign of this attribute in the main documentation entry for the ISP element, section 4.2.1. |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of the ISPs this element represents. Optional, default value is 1. |

4.2.21 FlexOfferResponse

Upon receiving and processing a FlexOffer message, the receiving implementation must reply with a FlexOfferResponse, indicating whether the flexibility offer was processed successfully.

XML representation summary

```
<FlexOfferResponse
  Metadata...
  FlexOfferMessageID      = UUID
  Result                   = ("Accepted" | "Rejected")
  RejectionReason          = String (Only if Result = "Rejected")
/>
```

| | |
|--------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexOfferMessageID | MessageID of the FlexOffer that has been accepted or rejected. |

| | |
|-----------------|--|
| Result | Indication whether the flex offer was accepted or rejected. Rejection is allowed in case no matching FlexRequest can be found, or the FlexOffer does not contain a valid price for each ISP with Disposition=Requested in the corresponding FlexRequest. |
| RejectionReason | In case the offer was rejected, this attribute must contain a human-readable description of the reason. |

4.2.22 FlexOfferRevocation

The FlexOfferRevocation message is used by the AGR to revoke a FlexOffer previously sent to a DSO. It voids the FlexOffer, even if its validity time has not yet expired. Revocation is not allowed for FlexOffers that already have associated accepted FlexOrders.

XML representation summary

```
<FlexOfferRevocation
  Metadata...
  FlexOfferMessageID           = UUID
/>
```

| | |
|--------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexOfferMessageID | MessageID of the FlexOffer message that is being revoked: this FlexOffer must have been accepted previously. |

4.2.23 FlexOfferRevocationResponse

Upon receiving and processing a FlexOfferRevocation message, the receiving implementation must reply with a FlexOfferRevocationResponse, indicating whether the revocation was handled successfully.

XML representation summary

```
<FlexOfferRevocationResponse
  Metadata...
  FlexOfferRevocationMessageID = UUID
  Result                       = ("Accepted" | "Rejected")
  RejectionReason              = String (Only if Result = "Rejected")
/>
```

| | |
|------------------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexOfferRevocationMessageID | MessageID of the FlexOffer that a revocation message was received for. |
| Result | Indication whether the revocation was accepted or rejected. Rejection is allowed in case the FlexOffer is unknown (it is the responsibility of the sending party not to revoke FlexOffer messages that have not yet been accepted), or the offered flexibility has already been ordered. |
| RejectionReason | In case the revocation was rejected, this attribute must contain a human-readable description of the reason. |

4.2.24 FlexOrder

FlexOrder messages are used by DSOs to purchase flexibility from an AGR based on a previous FlexOffer. A FlexOrder message contains a list of ISPs, with, for each ISP, the change in consumption or production to be realized by the AGR, and the accepted price to be paid by the DSO for this amount of flexibility. This ISP list should be copied from the FlexOffer message without modification: AGR implementations will (and must) reject FlexOrder messages where the ISP list is not exactly the same as offered.

XML representation summary

```

<FlexOrder
  Metadata...
  Period                = Period
  CongestionPoint       = EntityAddress
  FlexOfferMessageID    = UUID
  ContractID            = String (only if the offer and order refer to a
                           bilateral contract)
  ServiceType           = String (optional)
  D-PrognosisMessageID  = UUID (if present)
  BaselineReference     = String (if present)
  Price                 = CurrencyAmount
  Currency               = ISO4217Currency
  OrderReference        = String
  OptionReference       = String (only if the FlexOrder contained multiple
                           OfferOptions)
  ActivationFactor       = Number (optional [0.01-1.00])
  Unit                  = PowerUnitType (optional)
  <ISP
    Power                = Integer
    Start                = Integer
    Duration             = Integer (optional, default = 1)
  />
/>

```

| | |
|----------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| Period | Period the ISPs referenced in this FlexOrder message belong to. |
| CongestionPoint | Entity Address of the Congestion Point this FlexOrder message applies to |
| FlexOfferMessageID | MessageID of the FlexOffer message this order is based on. |
| ContractID | Reference to the bilateral contract in question, if applicable. |
| ServiceType | Service type for this request, the service type determines response characteristics such as latency or asset participation type. Each contract may specify multiple service-types. |
| D-PrognosisMessageID | MessageID of the D-Prognosis this order is based on, if present |
| BaselineReference | Identification of the baseline prognosis, if another baseline methodology is used than based on D-prognoses. |
| Price | The price for the flexibility ordered. Usually, the price should match the price of the related FlexOffer. |
| Currency | ISO 4217 code indicating the currency that applies to the price of the FlexOrder. |
| OrderReference | Order number assigned by the DSO originating the FlexOrder. To be stored by the AGR and used in the settlement phase. |
| OptionReference | The OptionReference from the OfferOption chosen from the FlexOffer. |
| ActivationFactor | The activation factor for this OfferOption. If this attribute is omitted, a default value of 1.00 must be assumed. Notes: The ActivationFactor must be greater than or equal to the MinActivationFactor in the OfferOption chosen from the FlexOffer. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| ISP | |
| Power | Power specified for this ISP (in the specified `Unit`). Also see the important notes about the sign of this attribute in the main documentation entry for the ISP element, section 4.2.1. |

| | |
|----------|--|
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of the ISPs this element represents. Optional, default value is 1. |

4.2.25 FlexOrderResponse

Upon receiving and processing a FlexOrder message, the receiving implementation must reply with a FlexOrderResponse, indicating whether the update was handled successfully.

XML representation summary

```
<FlexOrderResponse
  Metadata...
  FlexOrderMessageID      = UUID
  Result                   = ("Accepted" | "Rejected")
  RejectionReason          = String (Only if Result = "Rejected")
/>
```

| | |
|--------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexOrderMessageID | MessageID of the FlexOrder that has just been accepted or rejected. |
| Result | Indication whether the order was accepted or rejected. Rejection is only allowed in case the FlexOrder was already accepted previously, cannot be found, or does not exactly match the contents of the corresponding FlexOffer. |
| RejectionReason | In case the order was rejected, this attribute must contain a human-readable description of the reason. |

4.2.26 Metering

The Metering message is an optional message. The DSO will specify whether metering messages are required for a given program. If metering messages are used then the AGR must send metering messages, with one message sent per connection point per day. The metering messages must all be sent before the settlement can be performed. It is recommended to send the metering messages daily, once the metering data has been collected for the day.

XML representation summary

```
<Metering
  Metadata...
  Period          = Period
  EAN             = EntityAddress
  Currency        = ISO4217Currency (optional field, no longer used)
  <Profile
    ProfileType    = MeteringProfileEnum
    Unit           = MeteringUnitType
    <ISP
      Value        = Integer
      Start        = Integer
    />
  />
/>
```

The profile types to use will be agreed between the DSO and AGR and all agreed profile types must be included. It is recommended to include the Power profile and in a following major version the Power profile will become a mandatory profile.

Note: The power profile represents the average active power during ISP, considering both import and export energy.
 $Power = (ImportEnergy - ExportEnergy) * (60 / ISP\text{-}Length\text{-}Minutes)$. For example with a 15 minute ISP length we have a multiplier of 4, with a 30 minute ISP length we have a multiplier of 2.

| | |
|-------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2 |
| Period | The day that the metering data covers |
| EAN | The connection ID which this data refers to |
| Profile | A profile carries a sequence of ISPs with a defined type of metering data |
| ProfileType | A value from the ProfileType enum, one of Power, ImportEnergy, ExportEnergy, ImportMeterReading, ExportMeterReading |
| Unit | MeteringUnit type, use kW for Power and kWh for energy profile types. |
| ISP | |
| Start | Number of the ISP this element refers to. The first ISP of a day has number 1. |
| Value | Metering, energy or price value at the end of this ISP, in the designated profile units |

4.2.27 MeteringResponse

Upon receiving and processing a Metering message, the receiving implementation must reply with a MeteringResponse, indicating whether the update was handled successfully.

XML representation summary

```
<MeteringResponse
  Metadata...
  MeteringMessageID           = UUID
  Result                       = ("Accepted" | "Rejected")
  RejectionReason              = String (Only if Result = "Rejected")
/>
```

| | |
|-------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| MeteringMessageID | MessageID of the Metering message that has just been accepted or rejected. |
| Result | Indication whether the order was accepted or rejected. |
| RejectionReason | In case the metering was rejected, this attribute must contain a human-readable description of the reason. |

4.2.28 FlexSettlement

The FlexSettlement message is sent by DSOs on a regular basis (typically monthly) to AGRs, in order to initiate settlement. It includes a list of all FlexOrders placed by the originating party during the settlement period.

XML representation summary

```
<FlexSettlement
  Metadata...
  PeriodStart                  = Period
  PeriodEnd                    = Period
  Currency                     = ISO4217Currency
  Unit                         = PowerUnitType (optional)
  <FlexOrderSettlement        (0...n)
    Period                     = Period
    ContractID                 = String (only if the offer and order refer to
                                a bilateral contract)
```

```

D-PrognosisMessageID      = UUID(only if it has been agreed that the
                             baseline is based on D-prognoses)
BaselineReference          = String (only if another baseline methodology
                             is used)
OrderReference            = String
CongestionPoint           = EntityAddress
Price                    = CurrencyAmount
Penalty                  = CurrencyAmount (optional)
NetSettlement             = CurrencyAmount
<ISP                      (0...n)
  Start                  = Integer
  Duration               = Integer (optional, default = 1)
  BaselinePower          = Integer
  OrderedFlexPower       = Integer
  ActualPower            = Integer
  DeliveredFlexPower     = Integer
  PowerDeficiency        = Integer (optional)
/>
/>
<ContractSettlement      (0...n, only in case of bilateral contracts)
  ContractID             = Text
  <Period                 (0...n)
    Period               = Period
    <ISP                  (0...n)
      Start              = Integer
      Duration           = Integer (optional, default = 1)
      ReservedPower      = Integer
      RequestedPower     = Integer (optional, only if present)
      AvailablePower     = Integer (optional)
      OfferedPower       = Integer (optional, only if present)
      OrderedPower       = Integer (optional, only if present)
    />
  />
/>
/>

```

| | |
|---------------------|---|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| PeriodStart | First Period of the settlement period this message applies to. |
| PeriodEnd | Last Period of the settlement period this message applies to. |
| Currency | ISO 4217 code indicating the currency that applies to all amounts (flex price, penalty and net settlement) in this message. |
| Unit | The unit of Power that applies to the Power of the ISP's (optional, if not specified, assume that the Power is specified in Watts) |
| FlexOrderSettlement | |
| Period | Period the FlexOrder refers to. |
| ContractID | Reference to the concerning bilateral contract, if it is linked to it. |
| MessageID | MessageID of the Prognosis message (more specifically: the D-Prognosis) the FlexOrder is based on, if it has been agreed that the baseline is based on D-prognoses. |
| BaselineReference | Identification of the baseline prognosis, if another baseline methodology is used than based on D-prognoses. |
| OrderReference | Order reference assigned by the DSO when originating the FlexOrder. |

| | |
|--------------------|--|
| CongestionPoint | Entity Address of the Congestion Point the FlexOrder applies to. |
| Price | The price accepted for supplying the ordered amount of flexibility as per the referenced FlexOrder messages. |
| Penalty | Optional (default zero) penalty due a non-zero PowerDeficiency. |
| NetSettlement | Net settlement amount for this Period: Price minus Penalty |
| ISP | |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of ISPs this element represents. Optional, default value is 1. |
| BaselinePower | Power originally forecast (as per the referenced baseline) for this ISP (in the specified `Unit`). |
| OrderedFlexPower | Amount of flex power ordered (as per the referenced FlexOrder message) for this ISP (in the specified `Unit`). |
| ActualPower | Actual amount of power for this ISP (in the specified `Unit`), as measured/determined by the DSO and allocated to the AGR. |
| DeliveredFlexPower | Actual amount of flex power delivered for this ISP (in the specified `Unit`), as determined by the DSO. |
| PowerDeficiency | Optional (default zero) amount of flex power sold but not delivered for this ISP (in the specified `Unit`), as determined by the DSO. |
| ContractSettlement | |
| ContractID | Reference to the bilateral contract in question. |
| Period | List of periods containing reservations |
| Period | Period the settlement refers to. |
| ISP | |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of ISPs this element represents. Optional, default value is 1. |
| ReservedPower | Amount of flex power that has been reserved (and not released using a FlexReservationUpdate message) |
| RequestedPower | Amount of flex power that has been both reserved in advance and has been requested using a FlexRequest (i.e. the lowest amount of flex power for this ISP). If there was no FlexRequest, this field is omitted. |
| AvailablePower | Amount of flex power that is considered available based on the FlexRequest in question. In case RequestedPower=0, AvailablePower is defined so that the offered power is allowed to be between 0 and AvailablePower in terms of compliancy (see Appendix 3 for details). In case RequestedPower ≠ 0, AvailablePower is defined so that the offered power is allowed to exceed the amount of requested power up to AvailablePower. If this is relevant for settlement, the DSO can include this field. |
| OfferedPower | Amount of flex power that has been reserved in advance, requested using a FlexRequest and covered in an offer from the AGR. If there was no offer, this field is omitted. If there were multiple offers, only the one is considered that is most compliant (see Appendix 3 for details about compliancy). |
| OrderedPower | Amount of flex power that has been ordered using a FlexOrder message that was based on a FlexOffer, both linked to this contract. If there was no order, this field is omitted. |

4.2.29 FlexSettlementResponse

Upon receiving and processing a FlexSettlement message, the AGR must reply with a FlexSettlementResponse, indicating whether the initial message was handled successfully. When a FlexSettlement message is rejected, the DSO should consider all FlexOrderSettlement elements of that message related to potential dispute.

XML representation summary

```
<FlexSettlementResponse
```

```

Metadata...
FlexSettlementMessageID      = UUID
Result                       = ("Accepted" | "Rejected")
RejectionReason              = String (Only if Result = "Rejected")
<FlexOrderSettlementStatus   (0...n, only if Result = "Accepted")
  OrderReference              = String
  Disposition                 = ("Accepted" | "Disputed")
  DisputeReason               = String (Only if Disposition = "Disputed")
/>
<ContractSettlementStatus    (0...n, only if Result = "Accepted")
  ContractID                  = Text
  <Period                     (0...n)
    Period                    = Period
    <ISP                      (0...n)
      Start                   = Integer
      Duration                 = Integer (optional, default = 1)
      Disposition              = ("Accepted" | "Disputed")
      DisputeReason            = String (Only if Disposition = "Disputed")
    />
  />
/>
/>

```

| | |
|---------------------------|--|
| Metadata | The metadata for this message. For details, see section 4.2.2. |
| FlexSettlementMessageID | MessageID of the FlexSettlement that has just been accepted or rejected |
| Result | Indication whether the message was accepted or rejected. Rejection is only allowed in case a FlexSettlement covering a given period was already accepted previously, or contains invalid data. If one or more settlement items included in the message are not in accordance with the AGR's accounting, the message should still be accepted, and the dispute communicated using the appropriate FlexOrderSettlementStatus elements. |
| RejectionReason | In case the message was rejected, this attribute must contain a human-readable description of the reason. |
| FlexOrderSettlementStatus | Each FlexOrderSettlementStatus element indicates whether the settlement details about a FlexOrder, included in the corresponding FlexOrderSettlement element, are accepted or disputed by the AGR. |
| OrderReference | Order reference assigned by the DSO when originating the FlexOrder. |
| Disposition | Indication whether the AGR accepts the order settlement details provided by the DSO (and will invoice accordingly), or disputes these details. |
| DisputeReason | In case the order settlement was disputed, this attribute must contain a human-readable description of the reason. |
| ContractSettlementStatus | Each ContractSettlementStatus element indicates whether the settlement details about a bilateral contract, included in the corresponding ContractSettlement element, are accepted or disputed by the AGR. |
| ContractID | Reference to the concerning bilateral contract. |
| Period | List of all periods from the ContractSettlement, including possible lacking periods |
| Period | Period the settlement status refers to. |
| ISP | |
| Start | Number of the first ISP this element refers to. The first ISP of a day has number 1. |
| Duration | The number of ISPs this element represents. It is recommended to merge consecutive ISPs with the same disposition. |

| | |
|---------------|---|
| Disposition | Indication whether the AGR accepts the settlement details as provided by the DSO from the ISP(s) in question (and will invoice accordingly), or disputes these details. |
| DisputeReason | In case the settlement from the ISP(s) was disputed, this attribute must contain a human-readable description of the reason. |

Handling disputed settlement details is a manual process outside the UFTP scope.

Appendix 1 Glossary

| | |
|------------------|--|
| A-plan | Aggregator-plan, used as Aggregator's nomination |
| AGR | Aggregator. Role whose goal it is to maximize the value of flexibility, taking into account customer needs, economical optimization and grid capacity. |
| BRP | Balance Responsible Party |
| C&I | Commercial & Industry |
| Common Reference | Information about the Energy grid and the involved parties, which needs to be unambiguously available to all participating parties. |
| Congestion Point | Point in de grid where the grid capacity is not always sufficient to distribute the requested amount of energy |
| CRO | Common Reference Operator |
| D-prognosis | Prognosis regarding the Distribution of energy |
| DNO | Distribution Network Operator |
| DSO | Distribution System Operator |
| E-program | Energy Program – aggregated (daily) energy transactions of a BRP to be provided to the TSO |
| FlexOffer | Flexibility Offer, response to a Flexibility Request |
| FlexOrder | Flexibility Order, a response to a Flexibility Offer |
| FlexRequest | Flexibility Request |
| Grid | Network for the transport and distribution of energy |
| ISP | Imbalance Settlement Period |
| MBMA | Measure Before Measure After (baseline methodology) |
| MCM | Market-based Coordination Mechanism (USEF) |
| MDC | Meter Data Company |
| P&S | Privacy & Security |
| Prosumer | A consumer which is capable of producing energy as well |
| Settlement | Determining the energy production and consumption and used flexibility as preparation for the billing process. |
| Supplier | Has a contractual relationship with Prosumers to source, supply and invoice energy |
| TSO | Transmission System Operator |
| UFTP | USEF Flex Trading Protocol |
| USEF | Universal Smart Energy Framework |

Appendix 2

Rationale for information exchange in flexibility request

USEF specifies that when sending a flexibility request, the DSO provides the following information about expected congestion. This appendix provides the rationale.

- **Information about expected congestion is provided to all registered AGRs:** The reduction required must be made available to all AGRs operating in the specific congestion area, due to the DSO's legal obligation to treat all market parties in a non-discriminatory way. The grid topology, however, is considered to be confidential information. Therefore, USEF recommends that information about expected congestion is not provided freely to the entire market at national level.
- **Information about expected congestion includes the direction of the overload (production/consumption):** A grid overload at a congestion point can occur as a result of either too much production or too much consumption. To reduce overload where there is too much production, either the production must be reduced, or the consumption must be increased. For overload where there is too much consumption, this is the other way around (increase production or decrease consumption). The implication is that the congestion information provided must be clear about the direction of the overload.
- **Information about expected congestion includes the volume of reduction required.** While it is sufficient for the DSO to provide only information about the need for reduction, rather than the volume of reduction required for an ISP, it makes sense to provide more detailed information as this is likely to result in a more optimal use of flexibility and therefore better usage of the grid.
- **Information about expected congestion includes available grid capacity for other ISPs.** Providing insight on the available grid capacity for the ISPs without congestion gives AGRs additional information that can be used to determine their flexibility offers. In addition, it reduces the likelihood that the load shift provided by the AGRs will create a new peak that overloads the grid, and thus is likely to converge quicker, making the time needed to align a D-prognosis shorter. Whether or not this time is needed mainly depends on the number of congestion points a DSO has defined. On the downside, the more information provided by the DSO, the easier it is to get insight into the entire grid topology, especially for crucial points in the grid.

Appendix 3 Flex reservation mechanism

As introduced in Section 2.2.2, DSO's may want to contract a minimum amount of flexibility to be sure that it is available on request. The FlexOption (not part of UFTP) is a bilateral contract between DSO and AGR to reserve a specified amount of flexibility at a specific location (i.e. congestion point), for a specified time schedule and duration. In the day-ahead grid planning process, the DSO's forecast will indicate whether the contracted amount of flexibility is actually needed. If not, the flexibility can be used for other purposes. The DSO can send a FlexReservationUpdate message to signal the actual needs to the AGR. This appendix describes the purpose of this FlexReservationUpdate message and its impact on the contractual obligations and the actual FlexRequest that may follow in the validate phase.



Figure 4-2: Example flex reservation and FlexReservationUpdate, displayed in graphs

Figure 4-2 illustrates how ISPs 11 to 16 are part of a contract, where the AGR is required to reserve flexibility at ISPs 12 to 15 (downwards power only – for more information on direction and amount of power, see Section 4.2.1). The DSO was obliged to send a FlexReservationUpdate before each deadline, confirming the reserved power or partially releasing it. The power values from the updated reservation should always be equal to, or lower than, the (absolute) values from the original reservation but, since the contract is out-of-scope for UFTP, the DSO and AGR should maintain their own agreements about this.

After the FlexReservationUpdate, the AGR is obliged to keep the amount of flexibility available – even if no FlexRequest follows. After a FlexRequest is published, the AGR is obliged to produce a FlexOffer that meets the contract within the boundaries of the FlexRequest, where only the lowest reserved and requested power is valid for the contract. This is illustrated in Figure 4-3:

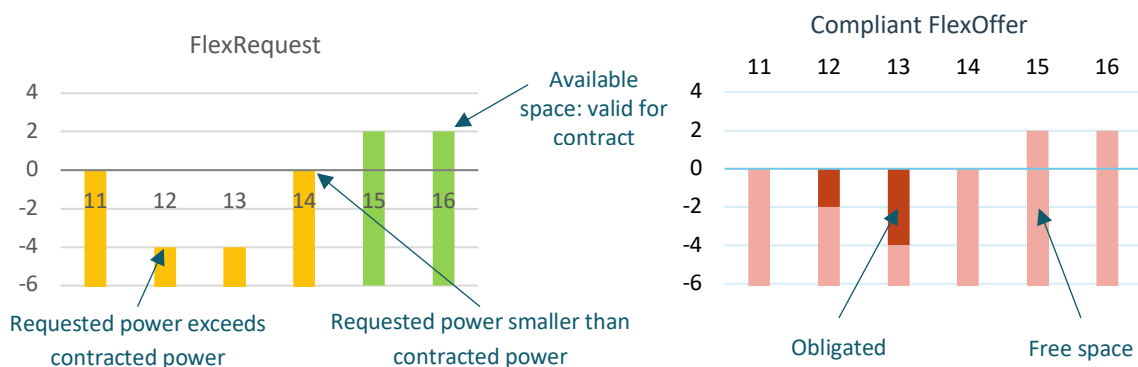


Figure 4-3: Example FlexRequest and requirements of FlexOffer, displayed in graphs

For further information on the FlexRequest, see Section 2.4.4.

Both the FlexReservationUpdate and the FlexRequest can only reduce the amount of power that the AGR is obliged to offer, as opposed to increase it. For ISP 12, the requested power exceeds the power that the AGR had to reserve in advance. The contract is therefore only valid for the lowest (absolute) power value, being -2. For ISP 14, the opposite is happening: the reserved power is not requested. Therefore, the AGR is not obliged to provide flexibility for ISP 14.

For ISP 15 and 16, available space is given for the AGR to deviate in the opposite direction. **For all ISPs that are included in the contract – even when the reservation is set to 0 – the AGR is obliged to conform to the restrictions given by the FlexRequest, including the bounds of available space.** The AGR is also free to provide more power than agreed, as long as it contributes to the decrease in congestion. In this example, the resulting FlexOffer that complies to the contract must be as illustrated in the graph on the right in Figure 4-3.

Appendix 4

Message transport mechanism

When implementing the USEF best-practice message transfer mechanism, the procedures outlined in this section apply.

1. Cryptographic Scheme (CS)

The USEF privacy and security guidelines require all participants to be able to securely transmit and authenticate messages. For these purposes, a transport-independent cryptographic scheme is specified, identified as Cryptographic Scheme Type 1 (CS1): implementation is a recommended practice for all implementations¹³. Based on NaCl¹⁴, a public domain library with high-speed state-of-the-art security features and a purpose-built and straightforward programming interface¹⁵, this scheme requires participants to generate two public/private key pairs:

| Purpose | NaCl function | Private key bits | Public key bits |
|----------------------------------|----------------------------------|------------------|-----------------|
| Digital signatures | <code>crypto_sign_keypair</code> | 512 | 256 |
| Authenticated message encryption | <code>crypto_box_keypair</code> | 256 | 256 |

The resulting private keys must be securely stored using an operating system or language-supplied facility for that specific purpose (e.g. the Win32 Data Protection API, OS X Keychain service or JVM KeyStore).

In addition to being stored locally, the public keys are exchanged with other USEF participants. For that purpose, the keys are joined together in a single 64-byte array (signature key followed by encryption key), Base-64 encoded and prefixed by the constant `cs1` and a period (indicating that the version 1 cryptographic scheme is used). For example:

```
cs1.V4lZrkYHq8FzneXxUML+QEMXMul3tBm+gPGoVDIZBA92VoWTu8/kRu2Zx72XOm1i/qwwoSgXjqSAqSa43myCaQ==
```

This public key string can be published in the DNS zone of the domain used in the associated entity address, or manually exchanged with other USEF participants. If DNS is used, implementation of DNSSEC is mandatory, as otherwise the key material would be completely untrusted. Implementations must use a DNSSEC validating resolver, which can either be built-in, or an external resolver on a fully trusted network.

When signing an outgoing message using the private digital signature key and NaCl's `crypto_sign` function, the result is an opaque blob: the sealed message. Unsealing this message, using the corresponding public digital signature key and `crypto_sign_open`, performs signature verification and returns the message plaintext. Since this all happens in the same API call, it is not possible to accidentally process an unsigned or incorrectly signed message.

Message exchange logic

The entire message exchange, from the client's outgoing message queue to the server's incoming message queue, using HTTP-over-TLS transport and the default cryptographic scheme, is visualized in detail in the picture below.

¹³ If this recommendation is not followed, a compatible implementation should be provided in order to achieve USEF compliance.

¹⁴ Pronounced "salt" and available with extensive documentation from <http://nacl.cr.yp.to/>. Sodium, <https://github.com/jedisct1/libsodium>, is an extended NaCl derivative with Windows, OS X and Linux platform support and many available language bindings. Sodium is highly recommended for USEF-compliant implementations.

¹⁵ Whereas typical cryptographic libraries require several steps to implement message encryption or signing, with each step opening the door for fatal programming mistakes, NaCl offers simple high-level functions which take care of everything, dramatically reducing the risk of inadvertent implementation errors.

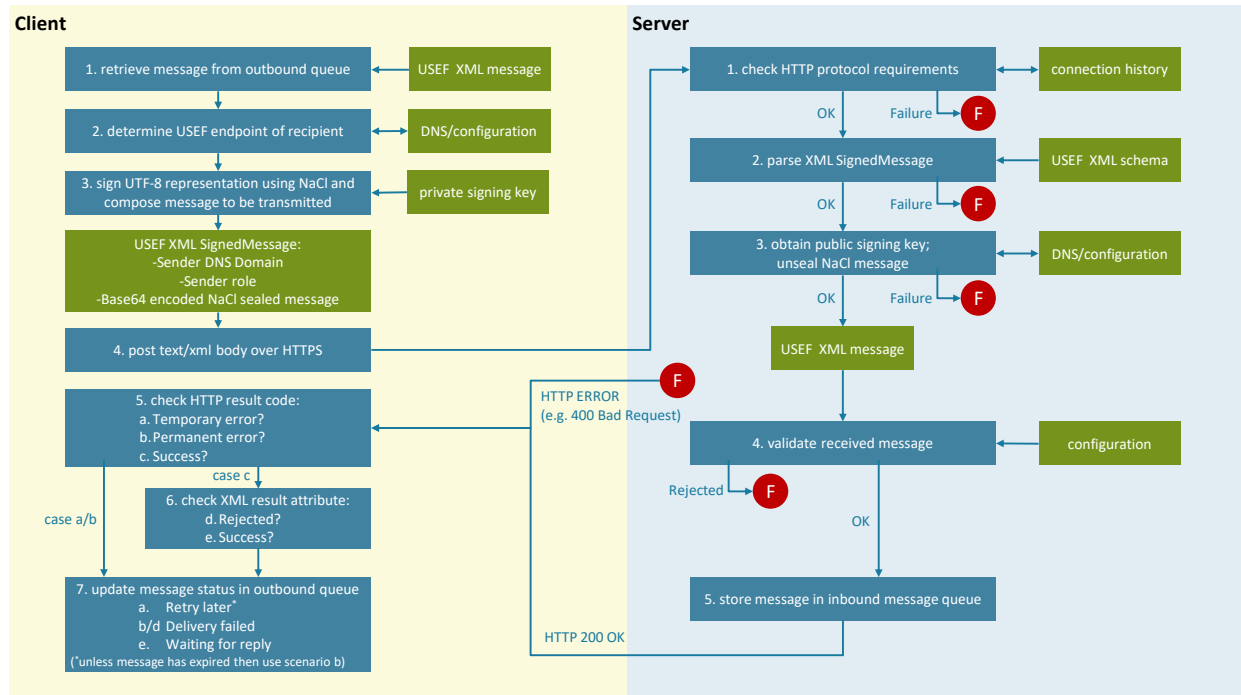


Figure 4-4: End-to-end USEF message exchange.

Messages go through the following stages between retrieval from the client’s outgoing message queue and delivery to the server’s inbound message queue:

2. Service discovery

Versioning

For versioning of the specification we use semantic versioning (see <https://semver.org/>). In short: MAJOR.MINOR.PATCH. Each new major version is incompatible with previous versions; New minor versions are backwards compatible with previous minor versions within the same major version.

Supporting multiple versions of the protocol

DSO’s should be more flexible than AGR’s, so DSO’s will support multiple versions of the protocol, but only for a restricted time. Also, the CRO should support multiple versions.

DSO’s and CRO are expected to support at most the three latest versions.

Version coupled to URL

Each major version should have a different url-path, so it is possible to route each incompatible protocol version to another service. This means that for minor changes and patches the url will remain the same.

The convention for the url-path (by example) is as follows:

version 2.0.0: /shapeshifter/api/v2/message

version 2.1.0: /shapeshifter/api/v2/message

version 3.0.1: /shapeshifter/api/v3/message

Messages

The metadata of each message contains the specification version.

A rejection reason will be added to reject a message with an invalid version

Service discovery

In the service discovery stage, DNS¹⁶ is used to discover the capabilities as well as the endpoint host name and IP address of the remote participant. Assuming the Internet domain name of the remote participant is `example.com`, the relevant DNS records are as follows:

| | |
|--|---|
| <code>_usef.example.com</code> | TXT record specifying the version of the USEF specification implemented by this participant, e.g. 2015 |
| <code>_role._usef.example.com</code> | TXT record containing up to two ¹⁷ space-separated Base-64 encoded public key strings for the specified role, where <i>role</i> is one of the values AGR, CRO or DSO. This data is only queried if (part of) the outgoing message needs to be encrypted |
| <code>_http._role._usef.example.com</code> | CNAME record indicating the HTTP endpoint receiving messages for the specified role. This label must not have any resource records of other types (with the exception of the mandatory DNSSEC-related records ¹⁸) and the alias should resolve to an A or AAAA record ¹⁹ . On the endpoint host, the implementation must listen on the well-known IANA-assigned TCP port for HTTP-over-TLS, i.e. 443. The implementation may listen on port 80 as well, but only for purposes such as supporting the HTTP/2 connection upgrade mechanism and never for unencrypted message exchange. |

The message will remain in the service discovery stage until all required data is available, or a participant-configurable and possibly message-class specific timeout timer expires. A DNS reply must have the NOERROR status, as well as syntactically valid and DNSSEC-authenticated content, in order to be considered usable. Other status codes such as SERVFAIL/NXDOMAIN, as well as connection failures or invalid record contents are considered temporary errors, in which case the lookup will be retried later within the timeout period.

To prevent transient Internet connectivity or participant configuration issues from inhibiting message exchange, the timer value used for such a period should be at least one hour for routine messages. If service discovery fails due to inability to collect all required data within this timeout period, the delivery of the message fails. Such messages are removed from the outgoing queue and returned to the sending process, indicating a transport error. Note that, although message queue implementations may inform the sending process about delivery delays, only the final success or failure status is authoritative.

When retrying failed requests, the implementation must implement an exponential back off mechanism to prevent undue load on the remote service. The implementation should set a reasonable maximum delay interval based on local factors, such as network capacity.

For test situations or small-scale deployments where service discovery via DNS is too cumbersome, or requires unreasonable amounts of additional work, participants should be able to specify service information in a local configuration file.

Since activities on the energy market are most likely regulated by national oversight bodies, and such organizations are natural candidates for running the authoritative DNS zone for accredited market participants, implementations are encouraged to support a list of DNS suffixes that correspond to the zones maintained by such regulators, and to automatically query these zones. For example, if an `energy.authority.test` is on the DNS suffix list, this zone is queried for `_usef.example.com.energy.authority.test` in order to determine if `example.com` is an authorized market participant.

¹⁶ Each USEF participant is responsible for publishing its own endpoint and public key information in a self-managed DNS zone. To prevent man-in-the-middle interference with the published information, use of DNSSEC is mandatory for such zones.

¹⁷ Usually one public key string will be present, but in order to support key rotation without invalidating in-transit messages, the implementation should list both the old and the new key for a short period of time (e.g. 24 hours) after a key has been replaced

¹⁸ USEF participants must enable DNSSEC and implementations must use a validating resolver, treating verification failures as temporary errors, eligible for retrying later. Absent DNSSEC authentication, private key strings must not be relied on, and separate manual secure key exchange is required.

¹⁹ As per IETF RFC 1034 [14], implementations must not fail when presented with CNAME chains or loops. Chains should be followed (to an implementation-defined maximum depth) and any loops or errors treated as temporary.

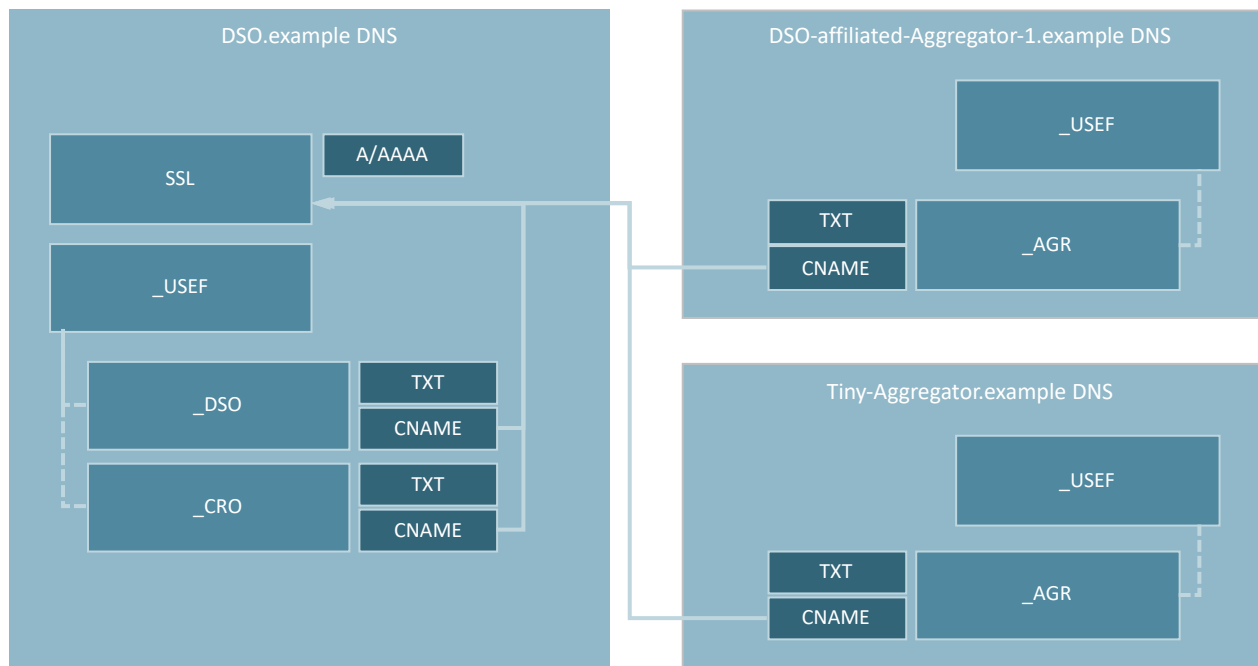


Figure 4-5 Example of the DNS relationships between a single USEF endpoint and multiple zones

An example of an anticipated DNS setup is shown in Figure 4-5: Here, an incumbent DSO (using the Internet domain `DSO.example`) operates a USEF endpoint, named `SSL.DSO.example`. The DNS entries for the USEF DSO and CRO role processes that underpin the DSO's core business both refer to this endpoint, ultimately leading to an IP/port combination usable for submitting USEF messages to.

In order to offer differentiated services to various customer segments, the DSO has also set up various AGRs, each operating its own web site and Internet domain. One of these AGRs is shown in the diagram. Instead of operating its own USEF endpoint, this AGR delegates this responsibility to its parent company, by pointing its USEF DNS records at the `SSL.DSO.example` endpoint.

The second AGR shown in the diagram is an independent market participant but lacks sufficient scale to warrant running its own USEF endpoint. Instead, it outsources that task to the DSO. Note that, due to the sensitive data included in some USEF messages, strong contractual and technological safeguards (such as full-message encryption) are required to correctly apply the USEF privacy and security guidelines.

Using a single USEF endpoint as shown here has a number of advantages, mostly related to the fact that only a single SSL certificate and IP address are required to operate the endpoint. Requiring unique instances of these would be complex and expensive, especially if the DSO runs tens or even hundreds of segment-specific AGRs.

There are, of course, disadvantages as well. Unless network-level measures are taken to prevent this, a single endpoint also means a single point of failure, and shared hosting scenarios that do not ultimately involve only a single legal entity require special attention to the protection of message contents.

3. Transmission

Once a valid service endpoint is available, the message enters the transmission stage²⁰. The implementation will now, for a reasonable amount of time (which is again at least one hour for routine messages, employing exponential back off), attempt to deliver the message to the remote participant.

²⁰ For messages containing non-public information, this stage transition may also include encryption of message sections using the public encryption key of the remote participant.

Messages are sent using a HTTP POST operation with the text/xml²¹ content type, the UTF8 character set and content-length indication. The request URI depends on the USEF implementation level and host name listed in DNS by the recipient. For USEF 2019 and the host `example.com`, it will be `https://example.com/USEF/2019/SignedMessage`.

Message content consists of a simple wrapper message, specified as SignedMessage in the USEF XML XSD, available for download from the public USEF web site²² at `https://usef.info/schema/2019` and documented in section 4.2

All usual protocol conventions should be followed during this stage. For example, when using HTTP version 1.1, redirects (responses with status code 3xx) should be honored in order to support load balancing. Any server errors (responses with status code 5xx), unknown response status codes, and connection timeouts and resets should be considered temporary failures and delivery should be re-attempted later within the timeout period. Only client success or failure messages (responses with status code 200 or a non-ambiguous²³ 4xx status code, respectively) should be considered final. This is standard HTTP 1.1 behavior, as fully described in IETF RFC 2616 [7].

4. Error handling

In the transmission stage, the guidelines for error handling are as follows:

Until the point that the message has been successfully validated against the USEF XML schema, errors are communicated using transport protocol mechanisms, e.g. HTTP status codes. For the HTTP protocol, the following error conditions are common (note that, depending on the server and libraries used, some of these may be handled automatically):

| Condition | Status code | Error type |
|--|----------------------------|---|
| Missing Content-Length header | 411 Length Required | Permanent (E cannot be resolved without changes to the sending implementation) |
| Invalid Content-Type header (not text/xml) or character set (default: utf-8) | 400 Bad Request | Permanent |
| Originating IP address is sending messages at a rate exceeding receiver policy | 429 Too Many Requests | Temporary |
| Body XML cannot be parsed or is not compliant with USEF XML schema | 400 Bad Request | Permanent |
| Failed to look up the sender's public signature key in DNS | 419 Authentication Timeout | Temporary (message can be processed when DNS is corrected/replicated) |
| Invalid signature (inner XML message cannot be unsealed, for example due to NaCl's <code>crypto_sign_open</code> function failing) | 401 Unauthorized | Permanent (since no authorization schemes that would allow the request to be resubmitted are available, unless private arrangements between sender and recipient exist) |
| Unsealed body XML cannot be parsed or is not compliant with the USEF XML schema | 400 Bad Request | Permanent |

Once a message has been determined to be valid USEF XML, errors are reported using a USEF XML response message with a successful transport disposition (e.g. HTTP status code 200 OK). Possible results in this stage are:

²¹ The USEF specification uses XML, since this format is already widely used in the rather conservative energy market, and enjoys wide and mature tooling support (particularly in the area of schema authoring and validation). Unlike other XML-based initiatives (such as WS-*), lightweight implementations are considered key, and alternate serialization formats (such as JSON) should be viable as well, despite being out of scope of this specification. If any such alternate message formats are implemented, fallback to XML must be provided as needed, or such implementations will not be USEF-compliant.

²² Please note that USEF makes no warranties whatsoever as to the uninterrupted availability of its web site, and that production services should not rely on the schema being hosted at this location for purposes such as validation: a local copy should be used instead.

²³ The prime example of an ambiguous HTTP status code is 404 Not Found: since this is commonly returned by front-end servers in case of temporary back-end issues, USEF implementations are encouraged to consider this a temporary error.

As in the service discovery stage, messages for which transmission permanently fails are removed from the outgoing queue and returned to the sending process, indicating a transport error.

Common Message Metadata

The USEF XML message obtained after successfully verifying the signature will have a root node corresponding to its message type, which must contain Metadata attributes common to all messages used by the implementation to process the message. The metadata attributes are defined in subsection 4.2.2

5. Awaiting reply

If message transmission succeeds, the implementation starts waiting for a reply from the remote service. Although replies to some messages can be expected immediately, other USEF messages may only elicit a response several hours later. In any case, responses are always decoupled from requests and implementations should be strictly asynchronous, correlating requests and responses using their conversation identifier.

Because of the asynchronous nature of message exchange, a process needs to be in place for handling messages that do not receive a (timely) reply. When preparing a new message, the implementation should specify the message class, as well as the time period within which a reply is expected. The following message classes are defined:

Incoming replies are validated using the steps outlined in the transmission stage. Where a reply is rejected, the pending outgoing message remains in the awaiting reply stage until it expires, and any additional manual processing is completed. Messages for which, ultimately, no reply is received are returned to the sending process, indicating expiry.

6. Service process message error handling

Since the message exchange process is responsible for transparently handling most, if not all, transient error conditions, service processes have no need for routine error recovery. In fact, any transport errors reported to the service layer are of such a severe nature that they most likely cannot be corrected without implementation or configuration changes:

- Malformed messages
- Entity address or signing key misconfigurationsExpired message, despite following manual recovery procedures for non-routine messages (for routine messages, no recovery is required by design but logging may definitely be appropriate)

The suggested error handling strategy for service processes is to mark the affected remote participant (i.e. not try to exchange any future messages with it) and to emit a diagnostic message indicating that manual intervention is required. After the issue is investigated and resolved, the configuration can be manually updated to re-enable the participant.

7. EA1 addressing scheme

In USEF messages, there is often a requirement for a globally unique identity for certain entities. To meet this requirement, USEF defines the Entity Address (EA). Each EA consists of a prefix, indicating the addressing scheme, followed by the actual address. Currently, two addressing schemes are supported:

- The European Article Number (EAN), commonly used to uniquely identify connection points in the electricity network and therefore a natural identifier to do the same in USEF. An example of an EA using this scheme is:
ean.871685900012636543
- The USEF type 1 entity address (EA1) is designed to allow participants to generate unique identifiers for themselves and entities managed by them, without relying on a central authority.

The USEF type 1 EA is structured analogous to the iSCSI IQN. Paraphrasing IETF RFCs 3720 and 3271 [8]: the EA does not define any new naming authorities but uses Internet domain names to ensure global uniqueness. Furthermore, the EA is constructed to give an organizational naming authority the flexibility to further subdivide the responsibility for name creation to subordinate naming authorities.

This makes the EA format slightly unwieldy but since bandwidth and storage space are not expected to be significant constraints in any USEF implementation, this is considered to be an acceptable tradeoff for not requiring a central authority.

Syntactically, an EA is a variable-length, 7-bit printable ASCII text string containing up to 255 characters. For example:

```
ea1.2013-11.info.usef.test:001:002.090807002a&b#
  <- naming authority -> < unique identifier >
```

Mostly taken from IETF RFC 3720 [9], below are the semantics of the various parts of this EA:

| | |
|----------------------------------|--|
| ea1 | Constant, indicating this is a type 1 USEF entity address |
| 2013-11 | A date code, in yyyy-mm format. This date must be a date during which the naming authority owned the domain name used in this format, and should be the first month in which the domain name was owned by this naming authority at 00:01 GMT of the first day of the month. This date code uses the Gregorian calendar. All four digits in the year must be present. Both digits of the month must be present, with January == "01" and December == "12". The dash must be included. |
| info.usef.test | The reversed domain name of the naming authority (person or organization) creating this entity address |
| 001:002.090807002a&b# | A locally unique string, which may contain product types, serial numbers, host identifiers, or software keys (specifically: it may include colons to separate organization boundaries). With the exception of the colon prefix, the owner of the domain name can assign everything after the reversed domain name as desired. It is the responsibility of the entity that is the naming authority to ensure that the names it assigns are worldwide unique. |

Appendix 5 Bibliography

- [1] USEF, "Flexibility Value Chain (update 2018)," 8 10 2018. [Online]. Available: https://www.usef.energy/app/uploads/2018/11/USEF-White-paper-Flexibility-Value-Chain-2018-version-1.0_Oct18.pdf.
- [2] USEF, "White paper: Flexibility Platforms," 2 november 2018. [Online]. Available: https://www.usef.energy/app/uploads/2018/11/USEF-White-Paper-Flexibility-Platforms-version-1.0_Nov2018.pdf.
- [3] USEF, "USEF - The FrameWork Specifications 2015," 2015. [Online].
- [4] USEF Foundation, "USEF: The Framework Explained," USEF Foundation, Arnhem, 2015.
- [5] USEF Foundation, "USEF: The Privacy and Security Guideline," USEF Foundation, Arnhem, 2015.
- [6] USEF, "Recommended practices and key considerations for a regulatory framework and market design on explicit Demand Response," 2017. [Online]. Available: <https://www.usef.energy/app/uploads/2017/09/Recommended-practices-for-DR-market-design-2.pdf>.
- [7] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, ""Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616," 1999.
- [8] Bakke, M., Hafner, J., Hufferd, J., Voruganti, K., and M. Krueger, ""Internet Small Computer Systems Interface (iSCSI) Naming and Discovery", RFC 3721," 2004.
- [9] Satran, J., Meth, K., Sapuntzakis, C., Chadalapaka, M., and E. Zeidner, ""Internet Small Computer Systems Interface (iSCSI)", RFC 3720," 2004.
- [10] USEF Design Team, "Release Details USEF Specification 2014:I.I.", USEF Foundation, Arnhem, 2014.
- [11] S. e. a. Tierney, "Pay-as-Bid vs. Uniform Pricing," Fortnightly Magazine, March 2008.
- [12] Wikipedia, "IEC 62056," 27 March 2014. [Online]. Available: http://en.wikipedia.org/wiki/IEC_62056. [Accessed 28 May 2014].
- [13] P. v. Oirsouw, Netten voor distributie van electriciteit, Arnhem: Phase to Phase, 2012.
- [14] P. Mockapetris, ""Domain names - concepts and facilities", STD 13, RFC 1034," 1987.