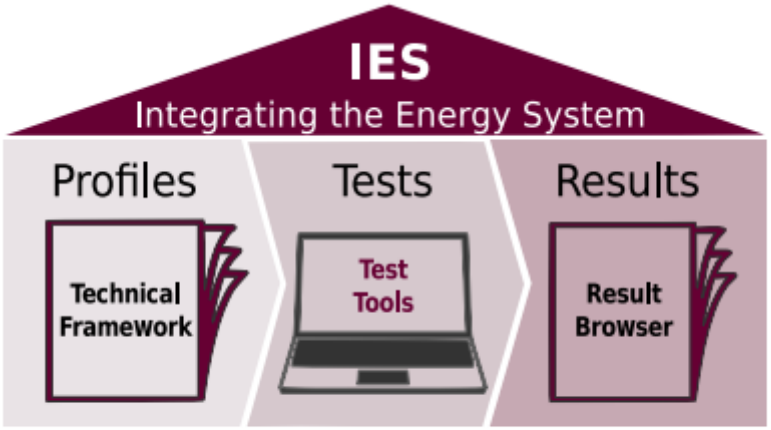
THE IES COOKBOOK

Enabling interoperability the IES way

Edition 0.9 – May 23, 2019



The project IES is part of the 'e!MISSION' program's 2nd call funded via the Austrian Climate and Energy Fund (KLIEN), managed by the Austrian Research Promotion Agency (FFG) under contract number 853693. IES is powered by the Climate & Energy Fund within the program 'Energy Research 2015'.

IES Integrating the Energy System

Document Title:	The IES Cookbook
Authors:	Gerald Franzl, Richard Pasteka, Marion Gottschalk
Version:	0.9
Date:	May 23, 2019
File name:	IES_cookbook.pdf

Change history

2019.01.21	official release for publication (print and distribute)
2019.01.28	publicly presented at Connectathon Energy 2019, Vienna, Austria
2019.05.23	extension by IES templates in the annex

This cookbook is intended for IES participants and addresses all aspects of the IES process, from agreeing on an interoperability issue to organising the annual Connectathon Energy event. The different focus is reflected by dedicated parts. It starts with Integration Profile Development because without profiles all other parts are meaningless. Where no own experience recommends amendments, the steps are based on established IHE practice (www.ihe.net) and common sense.

The IES-Austria project team are:

- Technology Platform Smart Grids Austria
  1060 Vienna, Austria
- Tiani Spirit GmbH
  1220 Vienna, Austria
- FH Technikum Wien
  1200 Vienna, Austria
- OFFIS e.V.
  26121 Oldenburg, Germany
- AICO EDV-Beratung GmbH
  2122 Ulrichskirchen, Austria
- Sprecher Automation GmbH
  4020 Linz, Austria

This IES cookbook is first published end of January 2019 in the course of the Connectathon Energy 2019, Vienna, Austria. It concludes the national funded IES Austria project. Currently, it has no ISBN or DOI assigned.[1] Alike all IES documents, the IES cookbook shall be a living document, that may be adjusted to consider feedback and changing environmental conditions. Being the first published version, this edition has trial status indicated by a version number less than one.

Table of contents

The IES Cookbook

Enabling interoperability the IES way

Edition 0.9 – May 23, 2019

Executive Summary

Abstract

Integrating the Energy System (IES) recognises interoperability as a key enabler for the deployment of smart energy systems, enabling new business options. Interoperability is covered in the SET-Plan activity A4-IA0-5 and the ISGAN Annex 6: Power Transmission & Distribution Systems.

Smart energy systems rely on trustworthy means to smoothly exchange digital information. Interoperable products enable heterogeneous systems and services based on components from any vendor offering solutions and devices tested according to the IES rules. Smart energy systems can be customised and adjusted on demand.

The stakeholder process proposed and maintained by IES [1], adopts the holistic IHE methodology standardised in ISO DTR 28380-1 [2]. This methodology evolves and spreads since 1999, driven by IT vendors from the health sector intending to achieve interoperable IT components for healthcare.

The brain of the approach are the Integration Profiles, which normatively state how to combine and implement established standards and good practice. The heart is the testing activity at an event called Connectathon Energy because it is about connectivity. The legs that shall get interoperable products to the market, are the Results Browser and the Integration Statements issued to state that a vendor offers compliant IT solutions.

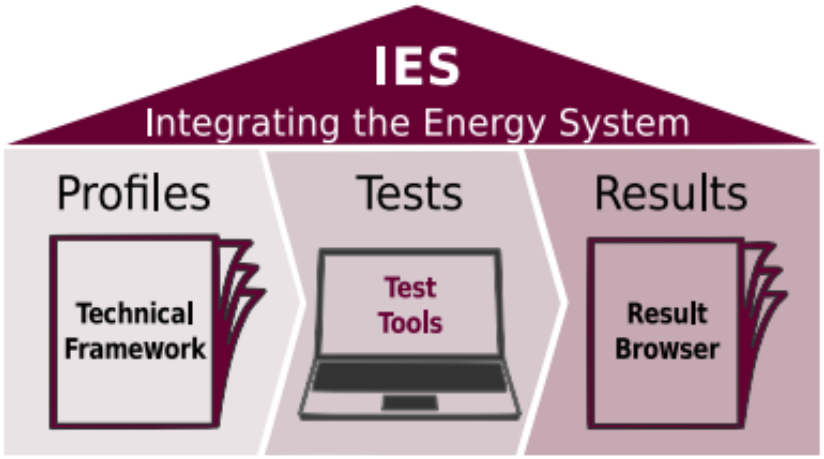


Figure 1: The three pillars of the IES methodology

These core parts constitute the three pillars shown in Figure 1. Together, they foster sustainable and efficient development and deployment of interoperable components for smart energy systems.

Finally, the creators of interoperable solutions and components are the hands that make interoperability flourish. In this cookbook, the steps and procedures of the IES process are outlined together with narrative explanation, reasoning, options and some background information.

IES workflow

The IES process is structured in the four basic steps shown in Figure 2, which split up into many intermediate steps and recursions presented in this cookbook.

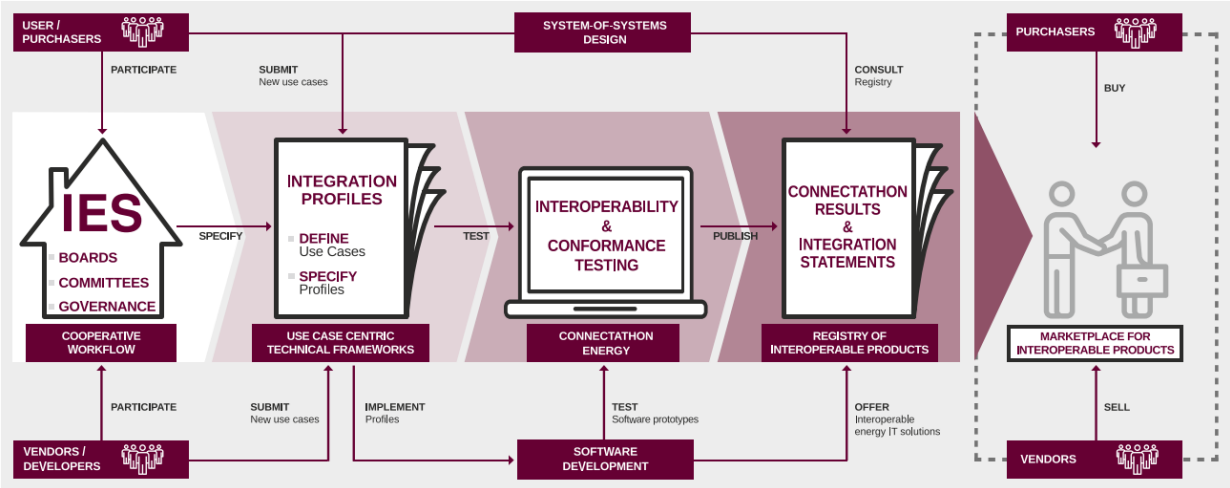


Figure 2: The IES process in four steps: identify - specify - test – sell

1. Identify Use Cases where interoperability is an issue and specify these by identifying system borders and requirements [3].
    1. Assign an interoperability issue to a domain (identify where the issue belongs to)
    2. Write a Business Overview (define actors, the environment and the general issue)
    3. Describe Business Functions (use the Use Case Method and UML use case diagrams)
    4. Reuse Integration Profiles where possible (save specification and test effort)
2. Jointly identify how interoperability issues can be prevented and specify the requirements normatively as Integration Profile [4].
    1. Evaluate which standards can be used to fulfil the Use Case requirements
    2. Specify the process to realise a Business Function (UML sequence diagram)
    3. Define the actors and transactions (decompose Meta-Actors into modules)
    4. Describe the role of the individual actors (modules)
    5. Draw an Actors-Transactions Diagram (visualise interaction)
    6. Draw detailed UML sequence diagrams per transaction (steps sequence)
    7. Specify additional communication and security requirements
3. Test independent prototype solutions against each other on annual plugfest and iteratively improve the Integration Profiles [5].
    1. Specify test cases and test sequences according to Integration Profile specification
    2. Add test cases, procedures, description and criteria to test environment (Gazelle [6])
    3. Create and integrate/implement conformity validation tools (e.g., Schematron)
    4. Develop and offer remote pre-Connectathon dummy test partners (optional)
    5. Execute test cases with at least two independent peer vendors
    6. Capture data transmitted during peer-to-peer communication (e.g., trace via proxy)
    7. Validate recorded messages/traces and log evaluated test results (impartial monitor)
4. Publish interoperability test results for each participant/vendor [5].
    1. Publish which vendors successfully tested an Integration Profile (Results Browser)
    2. Get written approval of interoperable implementation (Integration Statements)

Short narrative on process basics

Integration Profiles shall be living documents that are meant to be iteratively improved. Technical Frameworks shall continually grow to solve more and more issues of the business domain they cover. As

technology evolves, new Integration Profiles may complement the existing. Obsolete profiles remain available as reference for legacy system integration attempts.

The core idea of the IES methodology is agile cooperation between stakeholders: among users and technicians, scientists and engineers, managers, etc. All shall participate as peers and contribute jointly to the development of demand-oriented solutions. Interoperability may be achieved reliably with simple means that work fine for many.

The IES approach foresees that the implementer from different vendors test their solutions among each other, in a safe environment and in an early development stage.

All peers participating in a test case have a common goal: eventually they all want to pass the test. A multi-day plugfest provides the environment and time to track down errors and make corrections prior the decisive test. Implementer can talk to each other and jointly identify why something does not work as it should. Such issues are often based on different interpretation of the Integration Profiles, which demands understanding and amendment of the ambiguous text stating a requirement. The comments and errors recorded at the test event are most valuable to improve Integration Profiles. This feedback is practice driven and supports the advancement of the Integration Profiles.

Benefits – implicit and explicit

Public Integration Profiles yield increased development efficiency and market access:

- profiles provide clear answers to questions on possible options
- profile conformity allows small companies to offer sub-systems to integrate
- vendors need only one interface/solution to make their product interoperable

Contributing to Integration Profile development yields individual advantages:

- contributing parties can influence the solution design
- customers can make sure that profiles match their needs
- knowing specifications early enables foresighted development
- trust and respect among peers from working together on solutions

Testing solution prototypes at a Connectathon Energy yields these benefits:

- testing with peers helps to identify and solve interpretation problems early
- test partners help each other to pass the tests eventually (common goal)
- ambiguous specifications are jointly identified and reported for correction
- public testing success can convince customers (Results Browser – optional)
- profile compliance listing for products (Integration Statements – on demand)

Integration Statements optionally added to the public Products Registry:

- a shared, neutral, still valuable, marketing and advertisement option
- publicity for companies that launch products they might not be known for
- system purchasers can find matching components by comparing listed products

→ Integration Statements list the Integration Profiles a product supports. They become essential if tenders request certain profiles, which is a clear advantage for customers because they can precisely specify what fits into their infrastructure without going deep into technical details.

Framework structure

Technical Frameworks are a collection of documents according to the structure shown in Figure 3. They are developed strictly top-down, starting with a global Domain Overview (the business environment), followed by the Business Overview, outlining an application (business scenario) or service (business

segment). This Business Overview introduces the region covered by the Technical Framework at hand and is the first part to complete. For example, operating a VPP is in all possible variants a business use-case, whereas energy system security is a service that may be relevant for several applications.

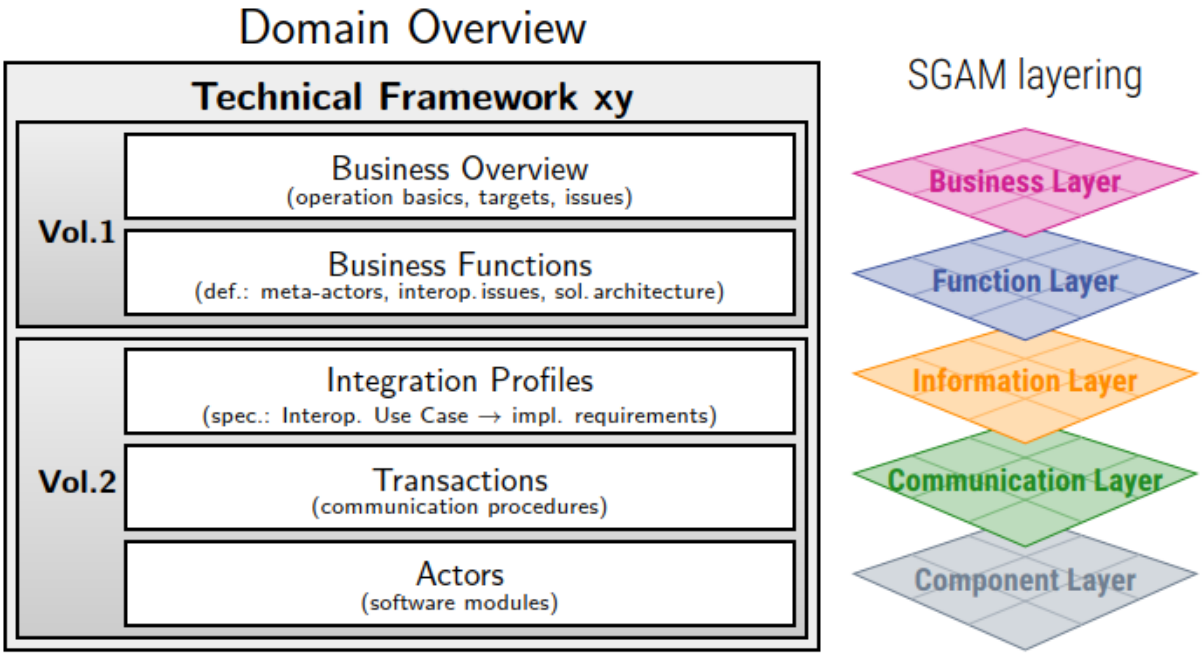


Figure 3: The IES Document Structure: roughly incorporating the five SGAM Layers [7]

Depending on the individual business design, different features are required. According to SGAM layering, we call them Business Functions. For example, telling a remote generator (DER asset) the production schedule for the next day, is a Business Function. Also predicting the demand schedule for the next day is a business function. However, if a Business Function does not involve cooperation with other entities, then it does not rise interoperability issues and needs no Integration Profile.

→ Incorrectly implemented calculations and data handling within an entity cause operational malfunction, but not necessarily an interoperability issue. Interoperability, as we interpret it, covers issues that reside in the cooperation.

Technical Frameworks consist of two basic Volumes, as shown in Figure 3. Volume 1 is purely informative and outlines the environment that the Integration Profiles from Volume 2 are addressing. Concluding the informative Volume 1, Actors-Transactions Diagrams visualise the situation, i.e., the required cooperation of the business entities (meta-actors) involved, and thereby indicate the Integration Profiles required. Volume 2 is a collection of normative Integration Profiles (specifications) and supportive information.

Amendment is required when new Integration Profiles become added to a Technical Framework. Integration Profiles specify a canonical solution for an issue that relates to a Business Function. Commonly, a new section and Actors-Transactions Diagram is added to Volume 1.

Process timing

The date of the annual Connectathon Energy determines the timing. The timeline in Figure 4 refers to the established practice from IHE.

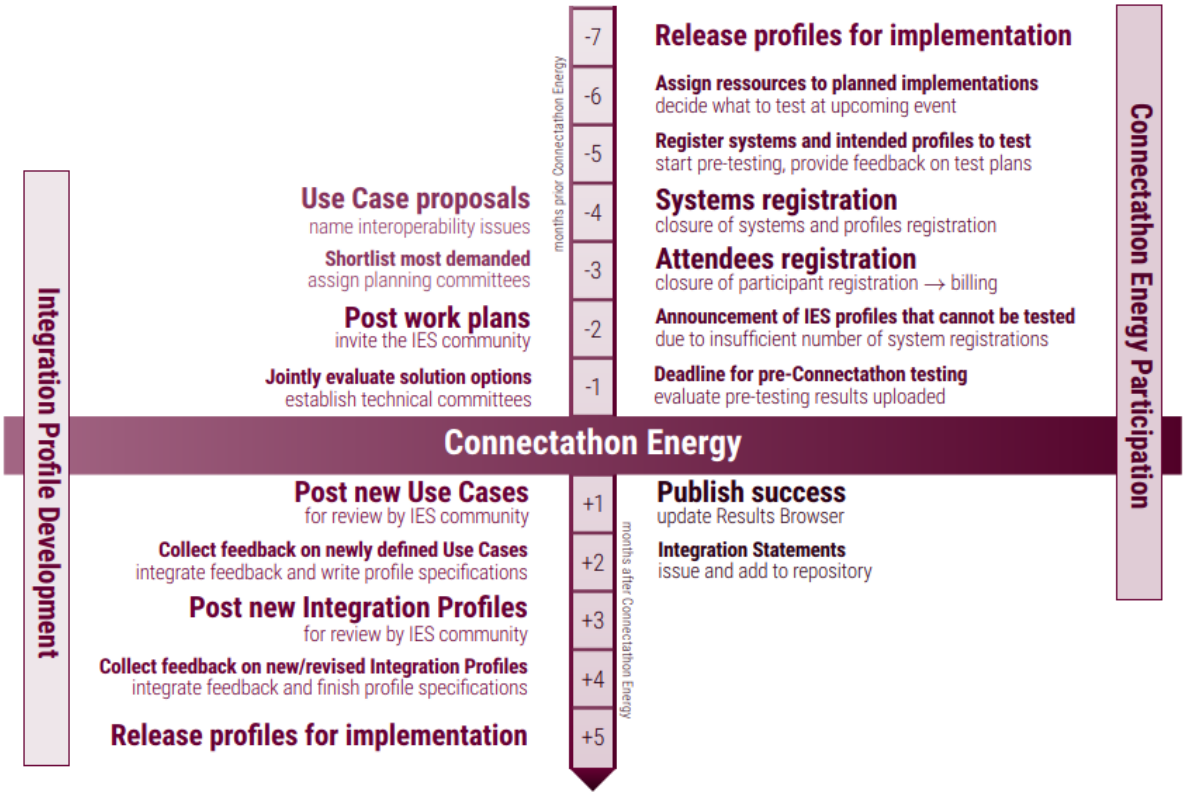


Figure 4: IES timing is centred on the annual Connectathon Energy

At first, the IES community collects and rates interoperability issues. When a core team of experts agreed to contribute towards solving one of these, the intended endeavour (work plan) shall be disclosed to the IES community two months in advance of the upcoming Connectathon Energy. Until the Connectathon Energy, the contributing partners survey ideas for solutions by collecting options, standards and good practice examples. Potential solution concepts for the most interesting profile ideas can be presented at Connectathon Energy side-events to engage and involve more experts.

One month later, the Use Case shall be completely formulated and posted for an open review. At this time the task force working on the specifications can go into specification details and shall post the resultant Integration Profile two months later, again for open review. Another two months later, the profile shall be published and is thereby announced ready for implementation.

If a sufficient number of Connectathon Energy participants register systems for testing the new profile, the testing becomes scheduled. Commonly, trial profiles needs to be revised based on the feedback collected. If only minor adjustments are needed, the profile becomes announced mature and is offered for regular testing and subsequent Integration Statements.

Independent whether a profile is new, revised, or stable, it can only be offered for testing at the Connectathon Energy if the required number and type of peers registered systems for testing a particular profile. However, unannounced ad-hoc testing of trial profiles may be possible, if the test plans are available on cite.

Crosscutting collaboration

IES focuses on the plurality that arises in the context of the system-of-systems interoperability challenge. Development teams working toward solutions need different skills and experiences to achieve the holistic IES targets. Figure 5 uses triples to express the plurality demand.

Figure 5: Plurality of desirable expertise in IES teams and committees

To consider { normation need – systems coordination – components architecture
domain coherence – application utility – implementation efficiency
testcase design – interoperability assessment – conformance validation

The IES process is intended to foster the development of interoperable solutions and services that increase the number and variety of products on the market, such that customised, and increasingly heterogeneous systems-of-systems can be composed.

IES offers the means to test early prototypes among peers in a safe environment outside the development lab, which clearly supports the development. Per profile certification achieves the same target but does not offer this add-on benefit.

Peer-to-peer tests do not qualify for issuing certificates. Therefore, interoperability certification is not replaced by IES because the former requires accredited test facilities to perform certification level testing. IES Integration Statements assert that the vendor successfully demonstrated a profile compliant implementation with the tested prototype or product.

Get in contact – get involved

Joining the IES community is easy, just express the wish to stay informed. News and participation opportunities will be distributed to the entire IES community. Wherever possible, feel free to address an IES representative to discuss interoperability issues, the process and visions.

IES Europe shall soon unite many IES activities. Until than, and aside from national initiatives that take over regional management tasks, the Technology Platform Smart Grids Austria lead by Dr. Angela Berger, will serve as contact point for all enquiries.

Dr. Angela Berger, Managing Director
Technology Platform Smart Grids Austria

Homepage: www.iesaustria.at
e-mail: ies@smartgrids.at

http://www.iesaustria.at

I. Integration Profile Development

The identification of an interoperability issue is the first step to be taken. Issues shall be thoroughly analysed and discussed among experts to avoid misclassification of problems that are not related to interoperability. For example: component malfunction, system misconfiguration, and environmental effects.

Integration Profiles are the normative parts of a Technical Framework. Their assignment can be done initially, prior writing the specification, but not before the Use Case is clearly specified. The process of adding and the subsequent management of Integration Profiles, Technical Frameworks and testing events, is presented in Part III.

Finally, note that here an existing Technical Framework is assumed and that the intention is to add specifications for a new Business Function identified to bear interoperability issues. That a capable group of experts participates in the endeavour is also presumed. How to start a new Technical Framework and how to best constitute a specification task force to write new Integration Profiles (a Technical Committee), are presented in Part III.

1. Explain the Business ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ application experts

State the application scenario where interoperability issues exist or may occur. Use a language that managers and technicians understand likewise.

1.1. State the interoperability demand

- **Identify the Business:** Explain setting characteristics and common variants
  → Add to Volume 1 Business Overview – integrate it if not already there.
- **Define generic Meta-Actors:** Identify actor roles and name them.
- **Identify Business Functions:** Explain interactions among Meta-Actors on the business level
  → Be generic: Make sure that the majority of business variants is covered by the identified Business Functions without requiring exceptions.
  → Add the result as new section at the end of Volume 1.

1.2. Identify Interoperability Use Cases

- **Name the features:** State what is (a) required to realise the Business Function, and (b) has potential for interoperability issues. Consider section 3 when specifying names.
  → Features that have no direct influence on exchanged data are considered to be Business Use Cases that need not be specified here.
  → Integration Profiles may solve isolated issues each, here called Interoperability Use Cases. Decomposing the Business Function reveals the isolated issues later on solved by one or more Integration Profiles.
- **Draw a Use-Case-Diagram:** Show how Meta-Actors are participating in Interoperability Use Cases (i.e., issues).
- **Draw an Actors-Transactions Diagram:** Show how Meta-Actors are connected by interactions (i.e., Transactions) to be specified in Volume 2.
  → Add these diagrams at the end of the new section in Volume 1.

2. Evaluate solution options ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ system architects

Best practice examples often guide the way to possible solutions for specific issues. If a plethora of Integration Profiles already exists, these represent best practice examples and shall be re-used where possible. Standards are commonly dedicated to an application field but may be migrated into new fields with minor amendments. Only if the options at hand are known, the best, most practical, least expensive solution for the Integration Profile can be chosen.

2.1. Define functional actors and transactions

→ Actor naming: Use self-explanatory names. Consider that send/request, set/get, and push/pull are synonyms used in different engineering disciplines. The implicit demand for interdisciplinary coordination prohibits a rigorously common language. Experts are assumed to know the different meanings that identical terms may have to different audiences and shall use concise language avoiding ambiguous terms.

→ SGAM interface between functional and information level.

- **Decompose Meta-Actors:** Define the roles of Meta-Actors in respect to the Interoperability Use Cases identified.
  → Identify the initiator and the responder of a communication, plus intermediate roles like relaying, aggregating, and logging.
- **Split business level transactions into components:** Commonly there are authorisation and authentication, action requests, reception confirmation, notifications, and possibly many more.
  → Drawing a flow-chart may be required to consider alternative paths and loops in case of complex communication patterns. Later on, sequence diagrams are drawn for each component. So here we need not go into the least bit.

2.2. State constraints and requirements

→ Knowledge about these is important to structure the subsequent specification efforts. However, this list can change and most likely will, until new Integration Profiles become mature.

→ SGAM interface between information and communication level.

- **Identify communication constraints:** For example, necessary trigger events, valid system state, conditional state machines, environmental conditions, and so on.
- **State all possible requirements:** For example, communication securing, access control, logging demand, etc.
  → If needed, decompose these as specified in 2.1. For example into: setup secure channel, close secure channel, connect to authentication service, request authorisation, open logging repository, log commands with user identity and timestamp, and so on. Most commonly, these add on functionalities are covered by existing Integration Profiles and established standards. The decomposition here is than required to specify the alignment of actors and transactions with those defined in the imported specifications.

2.3. Choose existing practices and international standards

- **Bundle existing Integration Profiles:** Compose a bundling table to identify the Integration Profiles (and CF) that are reused as integral part of the new Integration Profile. How they are actually included is specified with the transactions and actors definition. See Appendix B on the principle. Reused pieces of bundled Integration Profiles and CF are referenced and become adjusted to the needs of the here relevant Business Function.
  → Many of the identified requirements may haven been solved for some other Business Function already. Reusing profiles harmonises the solutions landscape and improves homogeneity within a system-of-systems.
- **Define the international standards to be used:** Decide which standards shall be used and specify the release (version/year) the specifications refer to. Provide a table, comparable to the bundling table, if different standards are required for the different Interoperability Use Cases covered in the Integration Profile.
  → Integration Profiles may cover any collection of issues, from a single Interoperability Use Case up to an entire Business Function. Applying the standardised Use Case Methodology recommends one profile per Interoperability Use Case. In practice, that may not be economic. Being living documents, profiles may be split into independent parts in the course of maturing.

- **Specify the best practice to be implemented:** If no existing profiles nor international standards can solve an issue, the solution needs to be specified within the profile. Some best practice solution becomes a de-facto industry standard for those that aim at profile conform implementation.
  → Specify the best practice (i.e., the technique) as detailed as possible. Be aware that profiles become open access documents ("Attribution – Share Alike 4.0 International" ⓒⓘⓢ). No copyright or intellectual property protected material may be enclosed in its entirety, these may be referenced and cited only.
  → Naturally, no proprietary protected non-free solution or part thereof qualifies for achieving interoperability in open heterogeneous environments.

3. Specify Integration Profile ○○○○○○○○○○○○○○○○○○○○○○ implementation experts

This endeavour follows the Use Case Methodology (IEC 62559) [8], and thereby uses the Smart Grid Architecture Model (SGAM) [7] [9] and The Open Group Architecture Framework (TOGAF) [10] to structure the process minimising the risk to overlook important requirements.

→ Use Case tools are often based on TOGAF for the sake of completeness. The V-model [11] is not so specific in the tasks and issue topics to consider, but also guides through the top-down approach. The more complex European Interoperability Reference Architecture (EIRA) [12] is based on a similar approach, and IHE standardised their top-down approach in ISO/TR 28380 [2].

→ For simplicity of understanding, without restricting a more efficient requirement and solution aggregation and decomposition outlined in Appendix B, henceforth singular comprehensive Integration Profiles are assumed, where all requirements and specifications are contained in a single linear document addressing a solitary Interoperability Use Case.

3.1. Define the Interoperability Use Case

- **Name it:** Specify an ID (alphanumeric code) and an explanatory name (text). This ID shall be the same as the indicator (and name) used in the Use-Case-Diagram and the Actors-Transactions Diagram in Volume 1.
- **Describe the interoperability issue:** Provide a narrative that concisely explains the issue solved, using terms and language understood by non-experts (end users, managers and directors, sales representatives).
- **Assign actors and transaction:** Define the actors and transactions that solve the issue if implemented according to the specifications stated later on.
  → Names only: Inserted here to gain an overview, and for quick access to where these are actually specified. Provide hyperlinks, if possible.

3.2. List standards and best practice

- **List standards:** Specify one-by-one the standards used to solve the Interoperability Use Case. Briefly validate the selection and state which parts are used for what purpose.
  → Include version/year and if possible, info on how to get access to the used standard. Briefly cite the essential parts required to implement the specifications stated later on.
- **List other sources:** Specify the best practice examples used to solve issues. Include a brief outline and a reference to your sources. If existing Integration Profiles get bundled, list them here as well.
  → If there is no accessible source, specify the practice in detail as far as required to implement the specifications stated later on. Possibly add a dedicated annexe and reference it.

3.3. Specify architecture building blocks

- **Decompose the Actors-Transactions diagram:** Provide a table listing all functional actors and transactions defined in 2.1, named in 3.1, and show how the Meta-Actors are composed

of these. For example, use an extended Actors-Transactions diagram showing the components of Meta-Actors and Transactions, to be specified later on.

- **Introduce functional actors:** Provide a brief narrative explaining in easy words what features the actor implements and what it is responsible for (requirements).
- **Introduce transactions:** Provide a brief narrative explaining in easy words what the transaction does and how that aim is achieved. Name involved actors, communication protocols used, and basic steps to be performed.
- **Actor role options:** State all the roles that Meta-Actors can have and relate them to the actors and transactions that the different roles require to perform the intended tasks.
  → Synonyms: List exchangeable role identification terms where supportive for better understanding among engineering disciplines.
- **Information flow, steps, sequences:** Explain how a transaction transfers information between the actors it involves. Draw a Sequence Diagram and explain each step. Define triggers and mandatory sequencing where needed.
  → The detailed specification follows in 3.4.
- **State technical constraints:** Provide a brief narrative explaining in simple words the communication channels to be used (e.g.: IP, Eth, LTE, PLC), message types and protocols to handle data (e.g.: TCP, HTTP, XML, MMS, ASN.1), operational bounds and limitations (e.g.: latencies, synchronisation, bit-error-rates), hardware interfaces, and whatever is required.
- **Explain security measures:** Provide a brief narrative explaining in simple words what is foreseen and required for sufficient prevention from malicious and accidental attacks (e.g.: access control, authentication and authorisation, data encryption, key management, access and message logging, required redundancy, fault detection, anomaly handling, fraud prevention).

3.4. Specify transactions

- **Scope of transaction:** Explain textually the interaction between two or more actors covered by the transaction specified. Include hints and exceptions where necessary.
  → Every sequence of actions that is step-by-step executed by different actors, represents a transaction to be specified. Commonly, each transaction covers a consecutive sequence of actions that for the related use case and business function cannot be split up into smaller parts reasonably. For example, the setup of a secure data connection consists of several sub-sequences, which could be specified as individual transactions (e.g., key exchange). For a use-case, the sub-sequence has no meaning if not embedded in the entire setup process. However, such sub-sequences may be imported from other Frameworks; e.g., a secure energy systems framework, which include a profile for private key management servers, for which the key exchange of two clients is a complete transaction.
- **Roles interaction:** List all involved actors in a table and briefly describe their role dependent interaction. If required, state into which Meat-Actors they shall be integrated.
  → Same as in subsection 3.3 but here with reversed viewpoint.
- **List applied standards:** Specify which standards are used and which parts and options thereof. Cite the essential features as much as required for correct implementation.
- **Draw Interaction Diagram:** Show the interaction that the transaction specifies by a UML sequence diagram.
  → Try to consider all sequence diagram aspects defined by the Object Management Group (www.omg.org) to seamlessly describe the data flow between each pair of actors.
- **Decompose the transaction into steps:** Based on the sequence diagram describe every step, one-by-one. Include for each step: intention, expected actions and responses. Use a flowchart to show alternative paths and loops.
- **Assign triggers, conditions and requirements to steps:** Specify triggers, conditions and requirements for each step.
- **Specify message semantics:** For each step describe and depict the composition of the message.

→ The message content may be encoded in a structured ASCII file or some data file attached to the message, a structured byte-block provided by the used message type, or is implicit to the message type (e.g., hello message, ping message).

- **Specify expected responses:** Describe possible results of the transaction; i.e., intended behaviour changes of actors in the course of the transaction and afterwards.
  → Possibly a response matrix is needed to cover all possible message contents and actor states that may occur.
- **Specify the security demand:** List technical, operational and legal requirements on the protection of exchanged information. E.g.: encryption strength and protocol to be used for some or all message contents.
- **Specify operational constraints:** List transaction features that need to be fulfilled for safe system operation. E.g.: worst case transaction latencies, process robustness, recovery performance, etc.

3.5. Specify actors

- **Scope of actor:** Explain textually the role of the actor. This should be rather simple because at this level of detail an actor should have only one primary task. For example, very basic, initiate communication and respond to communication.
  → Location less (virtual) actors: If necessary, an actor can be abstract. For example, a cloud repository or a swarm service. The integration of a distributed actor (its components) with other actors may cause additional requirements to be stated.
- **Specify message policy:** (a) For every message the actor can send, specify the conditions and pre-requisitions under which the message can be created and sent to specific other actors (roles). (b) For every message the actor may receive, specify the correct response. No action needs to be stated where applicable as the correct response.
  → The correct responses may depend on the received content and may be state dependent. The request-response 'matrix' can be multi-dimensional. However, these details are commonly specified by the used standard and need not be listed here if there is no corresponding ambiguity in the standard.
- **Specify the security demand:** List technical, operational and legal requirements on the availability and protection of actor features. E.g., access control, redundancy demand, intrusion detection, authentication, authorisation of communication partners, etc.
- **Specify operational constraints:** List actor features that need to be fulfilled for safe cooperation. E.g.: message buffer size, message loss probabilities, bit-error-rates, fault protection, resilience, robustness, availability, restart performance, etc.

4. Publish new specifications ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ domain experts

Once an Integration Profile has been written to the likes of all participating experts, it needs to be published and in consequence tested at an event called Connectathon Energy. Please refer to Part III and Part II on managing Integration Profiles and Testing at Connectathon Energy events respectively. Figure 4 in depicts the timing of these activities. Prior testing according prototypes need to be implemented, which the participating vendors most likely did already in parallel to specifying, to sort out technical specification errors early.

→ Note, IES profiles are public (i.e., CC BY 4.0) and it is in the best interest of all to attract more peers for early testing. Once knowledge about a specification is disclosed, anybody may implement it. IES does not manage the implementation.

→ Reference Implementations and Simulators: Requirements on developing so called Reference Implementations that guarantee to fulfil all specifications without any restrictions on functionality, and Simulators, being remote accessible dummy peers, required to perform pre-Connectathon testing, may be specified in a dedicated part or document to come.

4.1. Publish for trial testing

- **Hand over:** The implementation experts forward the new Integration Profile to the framework and domain management.
- **Assign to a Technical Framework:** If not already done, domain and application experts shall finally identify the framework the new profile fits into.
- **Formal check:** IES domain and application experts decide if a new profile is sober, is correctly assigned and conforms with IES policies and requirements.
- **Amend Volume 1:** The new Integration Profile may require changes in Volume 1 of the Technical Framework it is assigned to. Make sure the new profile is considered in the Actors-Transactions Diagram that links Volume 1 with the Integration Profiles specified in Volume 2.
- **Release for trial testing:** Application experts decide if the proposed specifications are sound and can be released for trial testing.
- **Trigger test preparation:** The release of a profile triggers the test preparation (step 5 in Part II) and the announcement of the new profile (task 10.1 in Part III).

4.2. Publish a revision – achieve maturity

- **Hand over:** The implementation experts forward the revised Integration Profile to the framework and domain management.
- **Formal check:** IES domain experts decide if the revised profile has sufficiently considered the feedback that made the revision necessary.
- **Release for mature testing:** Application experts decide if the specifications achieved maturity and either release it for mature testing or call for another trial testing cycle.
- **Trigger test adoption:** The release of a revision triggers the test preparation (step 5 in Part II) and announcement of the renewed profile (task 10.1 in Part III).

II. Integration Profile Testing

According to the V-model [11], test scenario and testing requirements shall be defined in parallel to decomposing and specifying the problem top-down. Testing at the plugfest called Connectathon refers to early integration testing, i.e., whether interfaces were compatibly implemented.

It is important to note that testing at a Connectathon does not replace the system verification and validation tests with or without the customer involved, nor any certification on standard conformance. For latter the interoperability tests are insufficient because they cover features required for a certain use-case only, not all the features specified in a standard. The former, system verification and validation, depends on the actual composition of a system-of-systems in the field. However, if the existing components were successfully tested, there exists a high potential that a new system that passed the same tests can be successfully integrated with less hassle.

→ *Test platform:* In the following we refer to the test platform Gazelle from IHE Europe [6]. This platform has been found feasible, but a different tool may be used similarly.

→ *Prototype testing:* When a system implements a new Integration Profile and is tested amongst peers for the first time, it is usually not a fully developed product. The implementer come to the event with their prototypes to identify shortcomings together with implementation and performance issues. The prime intentions are to identify misinterpretation of requirements early and to resolve them on site with the help of peers, at least long before product development is finished. It requires faith to go testing with peers having a prototype only, but it yields great rewards (win-win).

5. Prepare for testing ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ testing experts

Test scenarios and sequences are based on the use-case and the specifications defined in an Integrating Profile. In total, they shall cover all aspects specified, but no more.

→ *Negative tests:* Especially for security and safety requirements, it is important to design a sufficient number of negative tests. For example, tests that are passed if the system does not respond to erroneous and malicious requests. Not all possible attacks can be tested, it is therefore important to choose the most critical and those that cover a broad attack spectrum.

→ *Gazelle web-browser GUI:* All test definitions and reports shall be made available to the testing peers by Gazelle Test Management, in accordance to their role in the test.

5.1. Define test scenarios

- **Analyse Business Function:** First read the description of the Business Function that the Integration Profile relates to. Understand the demand on the business level.
  → Test scenarios shall be practice oriented.
- **Sketch typical communication:** Draw a sequence diagram for the test flow, being the sequence of test cases to be executed (or copy from profile if applicable).
  → Highlight if a test case prepares the environment for a subsequent test case; i.e., state whether the order matters.
- **Specify evaluation criteria:** Test requirements shall be clearly defined and the minimal requirements for successfully passing a test determined.
  → Highlight mandatory and optional test criteria, where latter exist.
- **State test environment constraints:** Textually describe the conditions and requirements that need to be fulfilled to execute the tests of a test scenario. For example, the required IT infrastructure and test case specific adjustments.
  → The content of messages exchanged over an end-to-end encrypted connection cannot be evaluated on profile conformance. Therefore, encryption possibly enforced by regulation in the filed, needs to be deactivated on purpose for certain test cases.

→ Testing whether an encrypted connection can be set up according to profile specification shall be specified in a dedicated test case. Where regional regulation demands encryption, these test cases are mandatory and have strict priority: no passing of any test case that in the field demands an encrypted connection, if setting up a secure connection according to specifications fails.

5.2. Specify test sequences and reports

- Analyse Integration Profile: See the specification of the Integration Profile to derive test cases and their steps. Each test case shall cover one feature, possibly in a certain system state. Each test case shall be split up into steps that challenge the communicating systems under test on certain aspects of the feature tested.
  → Draw flow charts and/or test tables to concisely define complex test sequences and multi-dimensional response checking respectively. I.e.: The correct response may depend on previous messages received and the systems' current state.
- Specify test communication sequence: For every test case specify the sequence of messages to be sent, received, and processed. Include optional and required acknowledgements as well as the correct answers/responses to requests.
  → Specify the content of test messages where it is required to determine if the receiving system correctly responds. For example, specify that in the first step a valid content shall be sent, in the following n steps invalid content that covers unacceptable content and not interpretable (incorrect formatted) content.
- Prepare a template to record and store test reports: Based on the defined test cases and test steps prepare a compact template to record test results. During test execution this template shall be provided by Gazelle's user interface.
  → Both, automated and manual recording of results is possible. Manual recording means that the test peers individually record the results by entering them via the Gazelle Test Management user interface (web-browser based GUI).

5.3. Create simulators (optional)

- **Virtual test peer:** Implement a software that mimics the behaviour of peer systems (roles) as far as required to perform trial conformity checks. The tool shall create dummy messages and responses similar to a real system but does not have to process messages contents.
  → *Dummy messages:* The created and responded messages need not be practically feasible in an operational sense. E.g., the same answer to the same type of message independent of the messages content is sufficient to verify that the message was received and logged for content validation.
  → *What matters is the ability to send, receive and respond to messages, to offer a remote sparring partner for pre-testing.*
- **Gazelle integration:** Simulators are configured and made available for testing. If configuration on the simulator side is required, e.g., port setting, then the simulator needs a web-browser GUI. In any case, the simulator is an individually addressable (connectable) software instance and shall be invoked independently for each peer connecting a prototype with it.
  → Independent instance: Each connection to a simulator shall be isolated from other connections to the same simulator (different instance).
- **Extension to simulated environment:** If for all sub-systems of a test scenario simulators are available, it might be possible to setup a simulated environment and replace one simulator by the system under test (i.e., hardware-in-the-loop testing).
  → To achieve a simulated environment, the simulators need to be more realistic models and implemented according to IES specifications. In that case, the simulators may be considered Reference Implementations of the according actors and roles.

5.4. Prepare validation tools

- **Reception tabbing or proxy server:** Either a proxy server is used to record all the messages exchanged between the systems under test in each direction, or the receiving peer needs to impartially log all received payload on behalves of the sending peer.
  → Recording at reception side needs trust, recording at the sender side is prune to cheating. Gazelle provides a proxy server that test-case specifically logs message-flows, such that multiple tests may be in progress simultaneously without interfering.
- **Message decoding and conversion:** The messages exchanged are encoded in a format (syntax) specified in the Integration Profile. To verify the context with Gazelle tools, these need to be converted into readable text (i.e., XML format).
  → *Readable XML:* If no automated content checking is available, the peers can manually check message contents by opening the XML file in the browser. Therefore, all data shall be converted into meaningful text, i.e., binary numbers into readable ASCII figures.
  → *Data Format Description Language (DFDL) is an Open Grid Forum proposed recommendation* (https://www.ogf.org/documents/GFD.207.pdf).
  → *Implicitly, conversion tests on correct encoding:* If the conversion into ASCII text fails or delivers unexpected results, the message was probably faulty encoded by the source.
- **Rules to determine correct content:** Specify the rules applicable on the contents of the XML files, e.g., a Schematron (https://en.wikipedia.org/wiki/Schematron). The rules result from the Integration Profile specifications and may be correlated from step to step.
  → *Message sequences:* To check whether a system's response is correct, the essential part of a transaction specified in the Integration Profile (i.e., the sequence of messages) needs to be checked on consistency. Multiple messages may therefore be converted into a single XML document.
- **Gazelle integration:** Validation tools that perform the message content verification can be integrated in Gazelle (as software tool or bidirectionally linked). Recorded message sequences can be sent via Gazelle to the validation tool and the verification result is automatically added to the test protocol.
  → *External tools:* Verification results from tools not integrated in Gazelle shall be copied into the report manually.

6. Testing at Connectathon Energy ○○○○○○○○○○○○○○○○○○○○○○○○○○○○ implementer

This part covers the steps an implementer is required to perform to successfully attend peer-to-peer interoperability testing at a Connectathon Energy event.

→ From system registration to executing test cases until test result validation by impartial monitors.

→ How to organise a Connectathon Energy event is covered in III section 10.

6.1. Testing event preparation

- **Registration:** To register, the peers need to express which actors and roles from which Integration Profiles they want to test at the event (system registration).
  → *Reference implementation:* Vendors with an already approved product may offer it to new peers attending with newly implemented components. However, reference implementations shall be open source alike all IES tools.
- **Test cases and peers announcement:** Shortly after the registration deadline, the test management team announces the testing schedule, containing scheduled times and dates, the test cases and the participating peers, based on the attendees wishes expressed with registration.
  → Only tests with sufficient number of registered peers are made available for testing.
  → Preferences on peers for testing are in general not allowed or considered.

- **Component testing:** If a simulator for the required communication partners is available, the implementer are advised to use them to test whether the messages they send conform with profile specifications.
  → Pre-event tests with simulators offer format (encoding) verification only. These are recommended to sort out basic errors prior connecting the prototypes with those of peers. At the event, problems that result from the combination of attending peers are in the testing focus, not malfunctions detectable without peers easily.
  → Attending the event with a components that are known to be not ready for testing, may prevent peers from having success at the event.

6.2. Execute peer-to-peer testing

- **Follow testing schedule:** Test scenarios and cases shall be performed with the foreseen peers according to the testing schedule announced by the test management.
  → The schedule assures that if all peers pass all tests, all peers are successful with the tests foreseen in the schedule. A sufficient number of repetitions with different peers is in all cases scheduled.
- **Execute test sequences:** The sequence of test cases and steps per test case are provided by Gazelle. In general, they may be executed in any order.
  → Note that sometimes the system state needed for test cases and steps may be bound to the previous tests. Either follow the recommended sequence or make sure that the system is ready to correctly execute tests.
- **Record test results:** In general, test results are to be manually confirmed by the peers participating in test cases. Recorded message flows are commonly automatically assigned and provided for verification tools. If otherwise instructed by the test case description, copy recorded messages into Gazelle prior invoking a verification tool.
  → *External verification:* If foreseen by the test case, use an external verification tool. How to perform the validation should be explained in the information on the test case within Gazelle.
- **Repeat test cases:** If something went wrong, i.e., a test or message verification failed, the peers can repeat test cases (steps) as often as needed to sort out the problem.
  → *System state:* When repeating test cases and steps be aware that some tests may require specific initial system states to be successfully executable.
  → In some cases, if possible at all, it may be wise to proceed in the sequence and find out what caused the problem afterwards. Seek help from your testing preers if you are not sure what went wrong. Your peers may face the same problem, so join ingenuity to track down the problem.
  → Arrange new testing time with your peers if testing shall be postponed to a later time because you need to make changes that cannot be performed on-the-fly.
- **Ad-hoc testing sessions:** Commonly, the planned testing schedule leaves gaps for on demand testing. This is arranged peer-to-peer on the floor and provides the option to find alternative peers to pass the planned tests.
  → Ad-hoc testing draft implementations in a state most likely not good enough to pass test cases is an option to check the current implementation among friends.

6.3. Get impartial success validation

→ To assure trustworthy test results, every test case needs to be validated by an unbiased monitor. The monitor is an expert that is not associated to any one of the participating peers. Monitors confirm successful testing, may give hints on possible problem sources, but cannot help in solving an issue.

- **Get a monitor:** If you are sure you passed a test case, then mark the test case in Gazelle being ready for validation. This automatically informs an eligible monitor. The monitor will than come to your place.
  → At the Connectathon Energy the monitors are located on a central table and can be easily identified by wearing a distinct piece of clothing. If no monitor shows up in due time, ask why.

- **Support the monitor:** Show the monitor the tests you performed. Monitors have their own web-based GUI and have access to all information recorded by the Gazelle testbed. If in doubt the monitor may request the repetition of a test case. Do so and explain for example why the results differ from the results expected in the test definition.
  → If you convince the monitor that the response is correct for the situation you test and does not challenge interoperability, then the monitor my confirm success (e.g., the test definition may be based on a different scenario).
- **Success verification:** If the monitor is convinced that all test steps of a test case and all test cases of a test scenario were successfully passed (according to specification), the monitor marks the test cases and scenario in Gazelle as verified.
  → *Feedback:* If peers or the monitor identify problems residing in the profile or test specifications, these shall be noted in the feedback option provided by Gazelle. Thereby, the feedback is destined directly to people responsible for the specifications.

7. Publish success ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ testing management

The team responsible for all testing, e.g., the Monitors, check the recorded results in Gazelle and awards the participants that successfully tested profiles.

7.1. Show success in Results Browser

- **Check test success:** If all tests related to an Integration Profile were passed successfully and with a sufficient number of independent peers, the attending peer is awarded a star for that Integration Profile.
  → *Exceptions:* The number of necessary repetitions may be adjusted in case the foreseen number of repetitions was not possible due to environmental conditions, e.g., a foreseen test partner cancelled attendance and there were no alternatives available on the floor.
- **Mark test accomplishments:** In the vendor vs. profiles matrix add a star in each field where a vendor successfully tested a profile at the event.
  → Testing prototypes does not allow to mention products here. Success at the Connectathon Energy states that the vendor has the knowledge to correctly implement the Integration Profile he was awarded for.
  → *Multiple years of successful testing adds to vendor reputation.*

7.2. Issue Integration Statements

- **List supported profiles:** A vendor can issue an Integration Statement for a product (name and version/year) that implements a successfully tested prototype.
  → To get an Integration Statement issued by IES officials, the vendor needs to come with the finished product to a Connectathon Energy and pass all tests again. This seems not foreseen in IHE and can be considered surplus if the vendor needs certifications that attest basically the same.
- **Dubious Integration Statements:** If IES fellows detect suspicious Integration Statements, they are encouraged to forward the information to the IES office.

7.3. Get listed in products repository

- **Public repository:** Integration Statements can be uploaded in a public accessible repository for broad visibility.
  → *Feedback:* Not only can system purchasers find compatible products from this list, they also may leave comments on the products if the vendor wishes to.
- **Maintain accuracy of repository:** Once a year, the vendors of listed product will be encouraged to remove products no more on sale. Vendors that vanished or do not respond, will be removed by the IES office.
  → There are no mature IES profiles yet, so there is no need for a repository yet. Its power comes with the number of products contained.

III. Integration Profile Management

What is developed and tested decides the IES community. Who is this community? All participating in both, the development of profiles and bringing products/prototypes for testing to the Connectathon Energy. Why? Because only if there is a sufficiently broad demand for some specification there will be enough peers that volunteer to write specifications and implement prototypes to perform interoperability tests among each other.

How Integration Profiles are developed and how their implementation is tested is outlined in Part I and II, respectively. What remains, is managing how the work performed by different teams of volunteers is embedded in a maximally self-maintained IES environment.

→ Note that the specified management tasks are based on visions, guesses, and practice copied from IHE. The IES management tasks shall be adaptive, i.e., community driven. Only the aim and the fundamental principles shall persist for sustainability reasons.

→ Management platform: Many of the required management features are provided by the test platform Gazelle from IHE Europe [6]. This platform has been found feasible, a different tool may be used similarly.

8. Profile proposal & revision ○○○○○○○○○○○○○○○○○○○○○○○○○○○○○○ IES community

If the number of independent peers wishing to test a new/revised profile is insufficient, the profile cannot be successfully tested and is withdrawn in task 10.1. Commonly, four independent peers need to bring prototypes or (sub-)systems supporting a new/revised profile to a Connectathon Energy for successful testing.

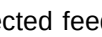
→ The three independent peers constraint is in place to implicitly prevent any proprietary solutions that would compromise the aim of IES for open interoperable systems-of-systems in the energy domain, not hindering any vendor from proposing an idea.

8.1. Establish a Technical Committee

- **Find development partners:** If the urge to solve an interoperability issue is sufficiently high it should be easy to find partners in the IES community with the same aim. Get in contact with peers and convince your company that it is economic to actively participate in joint solution specification.
  → *Benefits:* Early notion, joint forces, cooperation, learn from each other, support the solution you prefer. Refer to the Benefits section in the Executive Summary for details.
  → Currently, the IES community is small and finding peers is not as easy as it should be. Use your contacts and networks, address recent project partners and friendly competitors at fairs, events and conferences to increase the number of addressable peers. The IES management team advertises IES and organises complimentary events (as far as funding can be achieved) to support the IES community growth.
- **Organise cooperation:** Agile cooperation will result from the common aim if the committee covers all expertises required to specify a solution and is composed of peers willing to achieve a sustainable solution. To assure that, exclusion of distracting and non-supportive partners needs to be agreed and a priory clarified that an exclusion is no personal issue. Commonly, insufficient backing in the company causes peers to lack the required power and time to truly contribute. While working toward solutions, technical committees shall meet frequently. Meetings shall start with a brief individual progress report followed by open tasks to be addressed next (purely coordination). A shared document repository is encouraged to ease the information exchange.
  → IES timeline: The given dates, the development progress, and the support from team members determines the required meeting frequency. (cf. Executive Summary)

→ Telemeetings save time and costs, face-to-face meetings can be organised efficiently in the course of the Connectathon Energy. However, the kick-off meeting of newly established committees shall be face-to-face (if possible) to establish a social team spirit.

8.2. Evaluate a proposal

- **Publish in IES community:** As shown in the timeline presented in Figure 4, disclosing new profiles is a three steps task. First, the defined Interoperability Use Case shall be posted for review and comments from the IES community. Second, the specifications shall be posted for review and comments because only after the Use Case is clear the specifications can be holistically evaluated. In the third step, the new Integration Profile is published as open access document ().
  → Only after the collected feedback has been considered and integrated are the Use Case definition and the Integration Profile specifications made publicly visible. Publishing triggers the preparation of test plans and the implementation of new specifications by vendors from outside the IES community.
  → *Revision:* Integration Profiles shall be withdrawn from testing if a required correction is considered fundamental. Corrections in the specifications after publication always require to announce a revision and performing the complete profile development process.
  → *Minor corrections:* Grammar and writing issues may be corrected on-the-fly by authorised peers (e.g., the original authors) in the online provided documents according to the agreed versioning procedure, i.e., incrementing the sub-version index and stating the change and the author in the document history.
  → *Trial profiles:* If a new Integration Profile is first released, it is released for trial implementation only. In that case, the documents shall be conclusive, but need not have the rigour of mature profiles. Also, the timing is more flexible if the peers from the technical committee are sufficient to do successful testing among them. Note, Integration Statements may not be issued for Integration Profiles in trial state. Trial testing is performed aiming at testing the profile, not the preliminary implementation.
- **Collect feedback:** To eliminate minor faults and short-sighted specifications, both, new Use Case descriptions and Integration Profile specifications shall be posted for an open review process. Here, open means non blinded. Only registered users from the IES community shall see drafts and are allowed to comment. All comments and recommendations shall be signed; no anonymous access granted.
  → *Web-tool:* To efficiently collect feedback from the IES community a web-based sharing of drafts is encouraged. If the Gazelle platform [6] can provide this, it shall be used in general because it already includes user identification and eases coordination with test case development. If some other tool is used for test management, the registration and participant management features need to be provided similarly.
- **Evaluate:** Whether author or reviewer, focus on interoperability, i.e., the cooperation issues among systems. Are all aspects of interoperability covered? Are legal, semantic, syntactic, technical, and operational aspects considered? Also check for surplus confinement; the freedom of developers to choose how to implement features shall not be artificially constrained.[2]
  → Implementation details of individual actors shall not be restricted where not directly relevant for the realisation of the interaction specified for the Use Case.
- **Decision:** After the review and request for comments phase, the authors of specifications are empowered to decide by a qualified majority vote whether specifications are ready to be published for trial or mature testing.
  → It is the authors work and their blame if they decide wrong. They own the success versus failure balance and can take reasonable risks. Outsiders miss the intrinsic drive and may be improvident or overcautious depending on their personal risks only.
  → Decisions shall be strictly restricted to facts and the achieved feedback. Personal animosities and company policies shall not interfere.

[2] The exemplary profiles shall show the structuring only, they are not complete in the here claimed sense.

→ No feedback is also a kind of feedback, as are inappropriate feedback and phantom contributions intended to delay and obscure actions.
→ A team of experienced IES affiliates (e.g., domain experts) shall be empowered to sort out dubious contributions, to assess reasons for insufficient feedback, and to draw final decisions when requested by the authors or the community.

9. Profile assignment & release ○○○○○○○○○○○○○○○○○○○○○○○○○○○○ domain experts

As already mentioned, a committee composed of experienced IES participants shall supervise the activities and support the other teams. It is therefore slightly above the two others, as shown in the central part of Figure 5.

As outlined in Figure 4, the ideas for new profiles shall be shortlisted to focus on the most interesting and most needed. The experts proposing a new profile are evidently convinced that their idea most relevant. To make a fair selection, a neutral body shall hear the arguments and capture the market needs to rank proposals.

Based on the presented idea, the domain experts shall propose the reuse of existing profiles and decide which framework the new profile fits into. Once the framework is selected, the Planning Committee takes over. In case a new Technical Framework shall be started, the domain experts shall moderate the establishment of a new Planning Committee responsible for the new framework.

→ Every Integration Profile needs to be assigned to exactly one Technical Framework. Existing frameworks may be split into independent parts when convenient and technically reasonable.

→ Profiles shall be reused via bundling across frameworks wherever possible to minimise the specification demand and the solution plurality. Siblings of profiles shall be prevented.

9.1. Establish a Planning Committee

- **Find a diversified stakeholder team:** The initiators of a new profile idea shall in general be considered as members of the planning committee. However, coverage of relevant market sectors, envisaged application and system users and architects, and evidently the potential vendors, shall be assured.
  → *Roles:* A single committee member may cover several roles. This person than shall wear several hats and needs to switch roles in discussions, which may require exceptional skills when discussion is needed among roles represented by the same person.
  → In theory, a single person could constitute an entire committee, given this person provides the necessary skills and experiences, and is commonly accepted in this distinguished godlike position.
- Organise cooperation: Planning committees shall meet on a regular basis in alignment with the timeline presented in Figure 4. The proposers of new profiles, as well as any other community member raising a hand, shall be given sufficient time and attention to present their arguments. Decision shall be documented and openly available to the IES community.
  → *Moderation:* The Planning Committee has mostly monitoring and moderation tasks. It is responsible for the consistency of the framework as a whole, and in particular on Volume 1.
  → *Volume 1:* Committee members shall contribute an even share to Volume 1 to assure that all stakeholders are covered. Technical aspects shall be coordinated with the Technical Committees responsible for the profiles. In that respect, semantic consistency is a Planning Committee responsibility.

9.2. Support from Domain Coordination Committee

- Train and monitor committees: Senior members with a lot of experience concerning issues that may arise along the path (i.e., in the course of the entire IES process) are encouraged to share their knowledge and serve as guiding body represented in the Domain Coordination

Committee. The main task is to advise less experienced members and in particular the planning and technical committees.
→ Principles and norms: The Domain Coordination Committee is also the reference source to assure that the IES principles and norms are respected.

- Identify profile ambiguity and re-usability: Every community member is an expert on his own. The IES community is in general invited to point out any issues they detect in definitions and specifications. This feedback is the fuel that propels draft ideas into solid broadly accepted and respected specifications.
  → *Cross domain view:* In addition, commonly long before the entire IES community, the Domain Coordination Committee shall comment on ambiguities and possible re-usability of profiles when they are addressed to rank the ideas in subsection 8.2.
- Recommend framework re-organisation: Changing the structure of an existing framework may not be welcome by those that manage it. If the Domain Coordination Committee gets aware that strains toward reorganising condense, they shall support the Planning Committee throughout the change process.
  → *Change coaching:* Changes are a painful but often inevitable necessity. Lead from outside fewer personal issues need to be faced. However, make sure that all members of affected committees are involved, and their arguments taken serious from planning till implementation and evaluation. Only together the pain is spread and less of a burden.

10. Connectathon Energy organisation ○○○○○○○○○○○○○○○○○○○○○○○○ event experts

Whether big or small, a plugfest alike the Connectathon Energy requires preparation, from attendee management and support to infrastructure provisioning and general location logistics. The organisation quality has a considerable impact on the success and reputation of the event.

→ All timing is based on the IES timeline (Figure 4) and may be adjusted individually (i.e., per profile) if there is need and common acceptance from the involved peers, the test experts, and the event organisation team. Transparency is granted by public announcement in Gazelle.

→ Publicity: The Connectathon Energy events provide a good platform to publicly present the IES methodology toward better interoperability, the advantage of interoperable systems in general, and the profiles available now and in the near future. These aspects are primarily addressed by the side-events mentioned in subsection 10.3. In the focus of the IES cookbook on Integration Profile development, these activities represent dissemination and publication efforts, which today are required just as much as technical rigour.

10.1. Attendee management

- **Advertise profiles for testing:** According to the IES timeline, new profiles become advertised seven months prior the upcoming event. These shall be actively highlighted to raise attraction.
  → All profiles available in Gazelle (new, mature, and legacy) are implicitly advertised without further notice.
- **Register systems and profiles to be tested:** Until four months prior the event, vendors can register systems for being tested. All profiles available in Gazelle can be assigned for being tested with no constraints yet.
  → Specifying the profiles intended to be tested, is the core part of the system registration process and constitutes its necessity.
- **Attendance registration and billing:** One month after system registration closes, also the attendee registration closes. In general, each system registration includes two places (table space and chairs on the Connectathon Energy floor) to execute the registered tests in the course of the event. More places can be registered on demand.
  → Once systems and places are known, an invoice shall be issued and settled within common business terms prior the event. Only paid attendees get floor access clearance. On-site payment option may be provided only if the local infrastructure and organiser can handle it securely.

→ There is no on-site registration for the Connectathon Energy test-floor. However, on-site registration for side-events (10.3) may be offered on behalf of the organiser.

- **Announce profiles not to be tested:** Based on system registrations, i.e., the profiles intended for being tested, the event management identifies the profiles that cannot be tested due to an insufficient number of registered peers and informs the registered attendees until two months in advance of the event.
  → There is no need to announce which profiles will be tested, only those that were intended but cannot be executed need to be made public.
  → Informing affected attendees in advance offers them the opportunity to invite peers to add these profiles to their system registration (prior the deadline though).
- **Evaluate pre-Connectathon testing results:** Some test cases require successful testing of some features prior attending the event. These tests can be performed remote using Gazelle tools or external tools. The recorded or uploaded results need to be stored and verified in Gazelle until one month prior the event.

10.2. Event preparation and execution

- **Decide and announce upcoming Connectathon Energy venue:** Location and date of the next event shall be discussed and decided by the IES grandees (Domain Coordination Committee) and interested event organisers. Public announcement is achieved primarily via the IES homepage. Side events, i.e., the supporting programme, shall be announced by the event organiser on his or her behalf (see subsection 10.3).
  → The decision is best achieved in the course of the previous Connectathon Energy, i.e., one year in advance, such that it can be announced on site to all attendees and from there on in all media, i.e., the IES homepage.
- **Event infrastructure requirements:** Choose the venue such that it matches with the expected number of participants, provides all essential features, and supports somehow all other requirements. Required features that are not provided by the venue need to be contributed by the event organisers.
  → Essential features are:
    - Weatherproof room to operate sensible prototypes
    - Stable electricity with sufficient capacity
    - Trustworthy access restriction measures
    - Adequate restrooms (toilets)
    - Possibility to reliably receive and securely store equipment sent upfront

  → *Other features are:* Room heating and/or cooling, sufficiently powerful and stable Internet access, catering facilities, presentation rooms for side events, wardrobe service, meeting spots, quiet areas, and so on.

- **Setup test floor:** Setup the isolated test LAN connected to the Gazelle instance used to run all tests. Provide ample table space, minimum ~0.5 m² per booked floor place.
  → *Respect privacy of attendees:* Assure that attendees are grouped such that each vendor is an island where no competitor can peek into without being noticed.
  → *Assure security of the test LAN:* Assign a sufficient number of qualified network administrators to anticipate and fix issues prior serious harm is done.
- **Social event:** Aside from the political gala dinner organised by the IES grandees for their network of influencers and decision makers (10.3), the event organiser shall offer the test- floor attendees a private event toward the end of the plugfest to release some pressure and intensify fresh contacts. This may be combined with a short city-tour if offered by local authorities.
  → *Socialising:* The engineers participating in the test-floor shall be rewarded for the efforts contributed by attending the event in its whole duration, i.e., being available for peers to execute test cases, actively supporting more interoperability.
  → *What is remembered:* It is perfectly common that some tests fail when prototypes or trial profiles are addressed. A nice and private event among the attendees that worked with each

other for several days, is in any case a positive image they bring back home and remember when anticipating the next Connectathon Energy attendance.

10.3. Side events, catering and IES marketing

- **Organise side events:** Compose a small diversified program (single track) composed of presentations, panels and workshops on Integration Profile related topics for event visitors and decision makers that are not directly involved in the testing activities at the floor. Consider that the primary addressed auditorium are the executives of vendors and customers that IES wants to contribute to developing new IES profiles and to participate in peer-to-peer systems testing (→foster paid floor attendance).
  → *What is IES and how may IES evolve:* Present the basic idea and procedures, possibly followed by a panel discussion on improvement options and visions.
  → *New profiles:* Present the profiles that were developed recently and pending ideas for new profiles and amendments of existing profiles (e.g., as workshops).
  → *VIP tours:* Organise timed tours to the testing-floor for event visitors to show and explain the action on the floor. Keep some distance to regular attendees so the tours do not disturb them too much in their testing and bug solving endeavours.
  → *Conference/Symposium (optional):* Choose a hot topic and invite renown experts to talk about the future of the energy system, recent developments, fears and business opportunities. The conference/symposium shall be organised entirely by the event host at his or her discretion.
- **Catering and Gala dinner:** Organise adequate catering for the attendees of the floor and the visitors at the side events. While all floor attendees may mingle with event visitors, latter have no access to the floor centric catering.
  → *Catering at the floor:* Consider that floor attendees have work to do on individual schedules. Water, coffee, tee and snacks shall be provided at any time, and lunch over an extended time span, e.g., two hours plus. If a stationary venue catering facility (canteen) is available, lunch vouchers may be an option. No food on the floor, sensible systems shall no be jeopardised. Catering areas (as well as smoking areas) shall provide a utile space for private discussions among peers, e.g., tables, chairs, etc.
  → *Visitor catering:* Water, coffee and tee shall be offered throughout the day. Snacks and lunch shall be served as scheduled in the program. Light dishes cause less fatigue.
  → *Gala dinner:* A special event for the invited experts and executives from vendors and customers participating in IES and all the officers and functionaries visiting the event. If possible, get the dinner funded by a local vendor or authority. This event is most beneficial for the event host because he or she meets the important people. For the officers and functionaries it is a welcome favour that offers some networking opportunities.
- **Public relations and IES marketing:** The Connectathon Energy is the figurehead of IES. It is the most outstanding and visible signpost of IES besides the IES web page. The achievements at the Connectathon Energy fortify the IES process with evidence. Therefore, all marketing is concentrated on and around the Connectathon Energy.
  → The Connectathon Energy event itself is the most valuable IES advertising column and deserves an adequate and sustainable quality experience optimisation!

  In advance of the event:
  - Advertise the side events in an adequate form on trustworthy channels.
  - Use social media in a professional fashion to raise awareness (link to IES homepage).
  - Publish the program (agenda) as soon as possible via the IES homepage.

  During the event:
  - Assure that the signage at the venue sites fulfils all guidance needs.
  - Keep the on-line program up-to-date & advertise special highlights (reminders).
  - Say thank you to all participants, visitors, and the staff enabling the event.

  After the event:
  - Offer presentations and other presented digital content for download (asap).

- Analyse received feedback on recommended improvements → discuss & document.
- Conclude the event with a decent report published via the IES homepage.

IES supports open source because it is assumed to be the fastest way toward a wide spread knowledge distribution. No artificial barriers. However, there is high potential for misunderstanding because the approach is very different compared to common practice (i.e., certification). It takes time to change established habits.

→ Public relations and dissemination are the tools to foster comprehensive understanding of IES: the intention, the tools and processes, and the achievable results.

IV. Appendix

1. Document templates

Complimentary document templates are available for Volume 1, Volume 2, Integration Profiles, and Common Features. If support is required, get in contact with IES partners (ies@smartgrids.at).

→ https://www.smartgrids.at/integrating-the-energy-system-ies/technical-frameworks.html

In the following sub-sections, the content of the templates and an explanation are exemplary shown. Each template starts with a cover sheet, an explanation about the document, the document structure and a glossary. Moreover, each template ends with a reference list. For the layout of the templates please use the mentioned URL.

a. Volume 1

1. **About the Document**
   This section contains the document structure of Volume 1 and 2 of the Technical Framework.

2. **Definitions**
   At the beginning of the Technical Framework, technical terms are defined.

3. **Business Overview**
   Each technical framework has one business overview. This overview contains the description of a business that has interoperability issues like a Virtual Power Plant or Smart Metering. It is a textual description that can include graphics for a better understanding; additionally, a list with related standards and a short description can be given. Otherwise, no further guidelines are given for this section.

4. **Business Functions**
   The business overview contains at least one business function. At this point, an overview of the business functions is given through a short description. The complete business function is in an external file described with the IEC 62559 Use Case Template or created with a Use Case Management Repository (UCMR). Additionally, the Smart Grid Architecture Model (SGAM) plane can be used to locate the business functions to the domains and zones in the Smart Grid. So, the allocation in the electrical energy conversion chain and energy management processes takes place and gives a better understanding of the functions.

   4.1. Business Function
   For each business function, a new section is created that includes a textual description of the function and a UML Use Case diagram to show the involved actors, their relations to each other, and their functionalities within the business.
   Note: The business functions are described with the IEC 62559 Use Case Methodology; however, the complete Use Cases are stored in a UCMR and only the description and the Use Case diagram are part of the Technical Framework Vol. 1.

5. **Content of Volume 2**
   The informative view about the business case and functional description of the VPP is specified in this volume; the second volume of the Technical Framework includes the normative description of these with the IHE methodology. This includes the description of integration profiles and transactions, which specifies actors, security considerations, and data models for implementing the business function.

6. **References**
   All references used in the Technical Framework are mentioned here.

b. Volume 2

1. **About the Document**
   This section contains the document structure of Volume 1 and 2 of the Technical Framework.

2. **Definitions**
   At the beginning of the Technical Framework, technical terms are defined.

3. **Integration Profiles**
   In Volume 2, Integration Profiles are defined, and exemplary implementation options are provided by Transactions (aka Solution Building Blocks). Integration Profiles are normative descriptions of the features and specifics (Architecture Building Block Specification) that need to be implemented in order to realise the respective Use Cases (Architecture Building Block) in an interoperable manner. Commonly, where convenient, the name of the Integration Profile shall somehow reflect the name of the Use Case for ease of association. At this point, all operational Integration Profiles for the VPP are listed and linked to separate documents for the concrete specification. The basics for the implementation of the standards series IEC 61850 are shown in Section 4.

   3.1. <title of an Integration Profile>
   Each Integration Profile that is categorized to the Technical Framework is mentioned with a descriptive sentence and a link to the document, which includes the profile description.

4. **Implementation Strategy**
   The strategy of mentioned implementation options in the Integration Profiles shall be explained at this section. It is the basic for the implementation of the transactions in the profiles.

   4.1. General
   Firstly, an overview of technologies for the implementation strategy is given.

   4.2. Access Management
   Functional constraints, actor authentication, logging and other access control topics are described here.

   4.3. Security Considerations
   In this section, further mostly technical or legal requirements for the transaction are mentioned.

   4.4. Further information
   Additional information is given e.g. references to project websites or implementation examples. Furthermore, any kind of information shall be added here when it is relevant for the Integration Profiles.

5. **Abbreviations**
   Each abbreviation used in the technical framework are explained in this section.

6. **References**
   All references used in the Technical Framework are mentioned here.

c. Integration Profile

1. **About the Document**
   This section contains the document structure of Volume 1 and 2 of the Technical Framework.
2. **Definitions**
   At the beginning of the Technical Framework, technical terms are defined.

3. Integration Profile

In this document, an Integration Profile is defined, and exemplary implementation options are provided (aka Solution Building Blocks). Integration Profiles are normative descriptions of the features and specifics that need to be implemented in order to realise the respective Business Functions in an interoperable manner. Commonly, where convenient, the name of the Integration Profile shall somehow reflect the name of the Business Function for ease of association.

The following table specifies the mandatory actor grouping for the Integration Profiles defined in this document. Where possible, already defined and approved Integration Profiles from IHE are referenced. The IHE Integration Profiles can be found at http://www.ihe.net. For instance, the integration profile “IHE – Consistent Time” is mentioned.

3.1. Actors & Transactions

First, actors and transactions are shown in an Actors-Transactions Diagram, where dependencies between meta-actors and actors and the link to the transactions are visible. Second, all actors and transactions are listed in Table 2 with an optionality for the integration profile and a link to a brief explanation.

3.1.1. Actor Descriptions and Actor Profile Requirements

All actors involved (see Table 2) in the integration profile are described and profile requirements are mentioned. A reference to Section 3 of Volume 1 is possible.

3.1.2. Transactions

A list of all mentioned transactions in the Actors-Transactions Diagram with a brief explanation is given. The explanation shall contain actors involved, communication protocols needed and a summary of transaction steps.

3.2. Actor Options

Options and roles that may be selected for each meta-actor in this profile are listed in Table 3. The options like Station Controller, Plant Operator, Market participant etc. are described and dependencies between options are specified. The role specifies single actors of a meta-actor that are used in the transactions.

3.3. Information Flow Process

The transmission of data from client to server or the other way round is mostly achieved by a sequence of transaction steps. The detailed process flow and specification of steps, i.e., the transferred data objects and common data classes used per step, can be found in Section 4.

3.4. Communication Requirements

At this section general communication requirements for the integration profile are mentioned like the message type and transmission delay.

3.5. Security Considerations

At this section general security aspects for the integration profile are mentioned like the actor authentication and logging of messages.

4. Transactions

Transactions describe a concrete implementation of the interoperability issue described in the Business Function in Volume 1 of the Technical Framework which were specified through the Integration Profiles in Section 3. The interoperability issues are described in each transaction as brief interoperability use cases which demonstrate the challenge and actors involved of the transaction. Therefore, a UML sequence diagram is used to show single steps and the schema for the data exchange. Additionally, used standards and security considerations are mentioned.

4.1. Transaction

For each transaction, a subsection with a unique title is created.

4.1.1. Scope

The transaction description starts with a scope in which the interaction between two or more actors is textually described. Additionally, hints and exceptions can be mentioned.

4.1.2. Actor Roles

The actors involved in the transaction are listed in Table 4 with a brief description of the kind of interaction with the actors as well as the link to the meta-actor.

4.1.3. Referenced Standards

Standards which are used in the transaction are listed here. Optional, a short description of the standard can be included.

4.1.4. Interaction Diagrams

The interactions of the transaction are shown by a UML sequence diagram. It can include all aspects of the sequence diagram defined by the OMG to describe the data flow between actors. Based on the sequence diagram, single steps of the transaction are described by further subsections.

4.1.4.1. <title of a step>

Each step of the transaction is described by a new subsection by triggering events, message semantics and expected actions.

4.1.4.1.1. Trigger Events

Some conditions can be assigned to a triggering event which has happen before the message option is executed.

4.1.4.1.2. Message Semantics

The concrete message semantic is described and depicted. For instance, the scd file for the message exchange is demonstrated.

4.1.4.1.3. Expected Actions

Here the results and the behaviour of the actors after receiving the data are described.

4.1.5. Security Considerations

In this section, further mostly technical or legal requirements for the transaction are mentioned.

5. Abbreviations

Each abbreviation used in the technical framework are explained in this section.

6. References

All references used in the Technical Framework are mentioned here.

2. Flexible decomposition

Each Integration Profile solves primarily an interoperability issue of a targeted Business Function but may be reused via bundling it into an Integration Profile that contributes to solving other Business Functions. Graphically, this is shown in Figure 6, where also the option to further reduce specification redundancy via Common Features is considered.

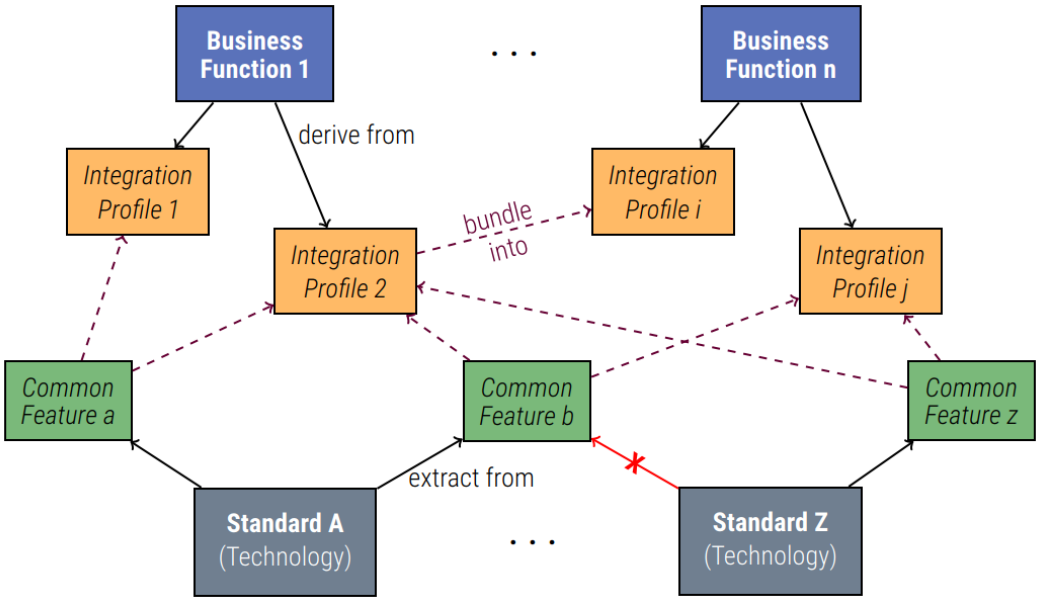


Figure 6: Profile Bundling and Common Features: avoid redundant specification

→ Common Features: Having noticed that modern standards offer a plurality of options to realise certain features, all based on the same technical background but with many options, it appears straightforward to use the features that a standard offers as solution building blocks. These can than be bundled (grouped) with Integration Profiles, as shown in Figure 6. Their actual implementation shall be constraint by the calling Integration Profile, such that interoperability is achieved precisely as required for the Business Function the Integration Profile relates to.

These solution building blocks we call a Common Feature (CF). They represent best practice solutions or excerpts from standards. They may refer to a single standard only and shall provide the full flexibility available. CF are economic if a CF is used by many Integration Profiles. In that case, CF save redundant specification of similar usages of the same feature. Modular profiles and CF may also be composed into complex holistic profiles using the approach presented in [13].

CF can be used to partially derive conformance tests. The more restrictive interoperability tests, and the conformance to the Use Case specific, Business Function related, restrictions, obligations, and constraints, cannot be derived from a CF because these are specified in the Integration Profile only.

3. Profile complexity recommendations

The complexity of Integration Profiles shall be limited to feasible specification and implementation effort. The IHE has formulated according recommendations and procedures that shall be applied likewise (https://wiki.ihe.net/index.php/Process).

→ IHE International Principles of Governance [14]

→ IHE Profile Design Principles and Conventions [15]

4. Certification approaches and examples

The IES scope is to establish a meaningful canonical, standardised process to create profiles for interfaces between components of Smart Grids. As of now, no such process existed in the domain of electrical engineering in the context of Smart Grids. There is no standardised way for system-of-systems based interoperability testing. However, a well-established concept exists, the IHE (Integrating the Healthcare Enterprise) approach.

The healthcare domain has similar problems motivated form the view of system-of-systems integration. However, it uses different processes, protocols, and data formats and ontology. A particular challenge is to establish these for the Smart Grid domain and to promote the well-established healthcare originated methodology in the energy domain.

Since this particular aspect is not in the scope of the EU and its European Interoperability Framework (EIF), going vertical with established methods, IES conducted the first Connectathon Energy together with IHE Europe in Den Hague, The Netherlands, successfully using the IHE Gazelle test platform. This first proof of concept was considered successful form the healthcare testing experts. Time will show how many interfaces and use cases will emerge with IES compliant profiles to be tested in the coming years.

→ *IES does not certify interoperability:* Testing at Connectathon Energy is among peers and on prototype level, i.e., in an early development stage. This does not qualify for certification level testing.

Interoperability certification is commonly offered specifically for a certain standard or technology. An example comparable in issues plurality are Bluetooth profiles (https://en.wikipedia.org/wiki/List_of_Bluetooth_profiles). The diverging plurality of possible requirements for different technical frameworks makes a global interoperability certificate unachievable.

Exemplary approaches and opportunities for certification and comparable assessments:

- IEC 61850 University: http://www.61850university.com/
- OpenADR: https://www.openadr.org/certification-process
- OPC foundation: https://opcfoundation.org/certification
- TÜV Süd Group: https://www.tuv-sud.com/industries/power-energy/smart-grid
- TÜV Rheinland: https://www.tuv.com/de/deutschland/gk/produktpruefung/smart_home_smart_grid_de/smart-home-smart-grid.html
- Smart Grid Interoperability Laboratory (SGIL): https://ec.europa.eu/jrc/en/research-facility/smart-grid-interoperability-laboratory
- EU Interoperability Centre: https://ec.europa.eu/jrc/en/research-facility/european-interoperability-centre-electric-vehicles-and-smart-grids

5. Definitions

- **Actor:** Is a functional software component of a system that executes transactions with other actors as defined in an Integration Profile.
- **Actors-Transactions Diagram:** Is the visualization of the relationship between (meta-)actors where the connecting transactions specify the interaction.
- **Business Case:** Is an economic viable realisation of a product/service idea/technique.
- **Business Function:** Is a comprehensible functional piece required to realise a Business Case.
- **Business Overview:** Is the narrative explanation of a Business Case including realisation variants and environmental aspects.
- **Committee:** Is a functional role within a community that is commonly taken by group of experts who perform the assigned tasks jointly.
- **Common Feature:** Is the specification of a single feature taken from a standard or good practice to be used (bundled) in Integration Profiles.
- **Conformance Testing:** Is a stand-alone process to ensure that the implementation conforms to specified standards and profiles, i.e., the implementation's outputs and responses are checked against patterns and rules.
- **Connectathon Energy:** Is a plugfest where components of different vendors are not hacked but connected to evaluate the interoperability among them.
- **Domain Overview:** Is the specification and explanation of the environment and constraints that identify and limit a business/market sector (domain).
- **Feedback:** Is a subjective information returned voluntarily. Recommendations, observations, comments, rating, grades, likes, hints, criticism, and praise are typical examples.

- **Integration Profile:** Is the specification required to realise a part of a Business Function (or combination thereof related to a single task) in an interoperable fashion (normalised).
- **Integration Statement:** Is the summary of the Integration Profile testing that the prototype modules of a specific product (family/version) successfully passed.
- **Interoperability [ITU-T Y.101, ITU-T M.60]:** Is the ability of two or more systems (applications, products, services) to exchange information and mutually make use of it.
- **Interoperability testing [ITU-T Z.450]:** Is assessing the ability of two or more systems to exchange information and to make mutual use of the information exchanged.
- **Interoperability Use Case:** Is a part of a Business Function that relies on data exchange between different actors (i.e., where interoperability is at risk).
- **Meta-Actor:** Is the composition of actors that joins all the functional components (actors) needed to integrate locally the functionalities (transactions) required for a Business Function.
- **Monitor (at Connectathon event):** Is a neutral person (testing expert) that verifies if a test case has been successfully passed by validating the recorded success.
- **Layer:** Is a functional grouping in the style of the generic ITU-T X.200 model. Connecting layers via interfaces enables adaptable system architectures.
- Peer-to-peer: Identifies non-hierarchical cooperation among equally empowered entities.
- **Technical Framework:** Is the hierarchy of documents that introduce, define and specify how to implement functionalities and features such that interoperability is achieved.
- **Prototype testing:** Is the testing of implementations in a state where adjustments and amendments can be directly integrated right away.
- **Sequence Diagram:** Is a UML conform interaction visualisation that shows different processes or objects (actors) as parallel vertical lines, and the messages exchanged in order of occurrence as horizontal arrows (time progresses top to bottom).
- **Simulator:** Is a virtual test peer that provide the essential features to perform conformance tests. Typically used for pre-Connectathon tests but insufficient to assess interoperability.
- **System-of-systems:** Is a system that results from the cooperation of different systems without implicit central control. Typically, a complex of individual control systems and diverse objectives.
- **Transaction:** Is the specification of a set of messages (1..n) exchanged among a group of actors that realise the Use Case specific information exchange (in one or both directions, in a strict or loose order) as specified by an Integration Profile.
- **Use Case:** Is a list of actions or event steps required to achieve a distinct goal, typically defining the interactions between a role (an actor in UML terminology) and a system. The actor can be a human or some other external system.
- **Use Case Methodology:** Is the approach to identifying and exchangeable documenting requirements based on TOGAF as specified in IEC 62559.
- **Use-Case-Diagram:** Is the UML conform visualization of a Use Case showing the relationship between users and the different use cases in which they are involved.
- **Validate:** Is to officially prove/state that something is true or correct.
- **Verify:** Is to show or agree that something is true or correct.
- **Volume:** Is used to group different components of a Technical Framework such that different experts can focus on different Volumes.

6. Abbreviations

ADR	Automated Demand Response
ASCII	American Standard Code for Information Interchange
ASN	Abstract Syntax Notation
CF	Common Feature
DER	Distributed Energy Resource
DFDL	Data Format Description Language
DLNA	Digital Living Network Alliance
EIRA	European Interoperability Reference Architecture
Eth	Ethernet

GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
ID	Identifier
IEC	International Electrotechnical Commission
IES	Integrating the Energy Systems
IHE	Integrating the Healthcare Enterprise
IP	Internet Protocol
ISO	International Standardization Organization
IT	Information Technology
LAN	Local Area Network
LTE	Long Term Evolution
MMS	Manufacturing Message Specification
OPC	Open Platform Communications
PLC	Power Line Communication
SET	Strategic Energy Technology
SGAM	Smart Grid Architecture Model
TCP	Transport Control Protocol
TOGAF	The Open Group Architecture Framework
UML	Unified Modelling Language
VPP	Virtual Power Plant
XML	Extensible Mark-up Language

Acknowledgements

The tremendous effort made by the members of the IES Austria project team, who participated to their best in nearly thirty workshop meetings, periodic teleconferences and on-demand communication and the valuable contribution of perfectly complementary expertises, are acknowledged with thanks. The significant contributions made by individual researcher and teams, the organisation efforts of the project lead and co-lead and the numerous contacts greatly promoted the approach in the energy systems community, which is to be acknowledged likewise. Finally, the innovators that participate with personnel at the first Connectathon Energy events earn our highest respect for bravery and endurance, and many thanks for the valuable hints and feedback.

List of Figures

References

[1] M. Gottschalk, G. Franzl, M. Frohner, R. Pasteka and M. Uslar, "From integration profiles to interoperability testing for smart energy systems at connectathon energy," *Energies,* vol. 11, p. 3375, December 2018.

[2] *ISO DTR 28380-1: Health Informatics - IHE Global Standards Adoption - Part 1: Process.*

[3] M. Gottschalk, M. Uslar and C. Delfs, "The Use Case and Smart Grid Architecture Model Approach: The IEC 62559-2 Use Case Template and the SGAM applied in various domains," in *SpringerBriefs in Energy*, Springer, 2017.

[4] M. Frohner, M. Gottschalk, G. Franzl, R. Pasteka, M. Uslar and S. Sauermann, "Smart grid interoperability profiles development," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2017, pp. 189-194.

[5] IHE Europe, "The IHE Connectathon. What is it? How is it done? Version 006," 2018. [Online]. Available: https://www.ihe-europe.net/sites/default/files/WP_Connectathon_2019_0.pdf.

[6] IHE International, "Gazelle - eHealth test framework for interoperability," [Online]. Available: https://gazelle.ihe.net/.

[7] CEN-CENELEC-ETSI Smart Grid Coordination Group, "First Set of Standards," 2012.

[8] I. 62559-2:2015, *Use case methodology – Part 2: Definition of the templates for use cases, actor list and requirements list,* 2015.

[9] SG-CG/RA, "Smart Grid Reference Architecture," CEN-CENELEC-ETSI Smart Grid Coordination Group, 2012.

[10] T. O. Group, "TOGAF Version 9 - The Open Group Architecture Framework (TOGAF)," 2009. [Online]. Available: https://www.opengroup.org/togaf.

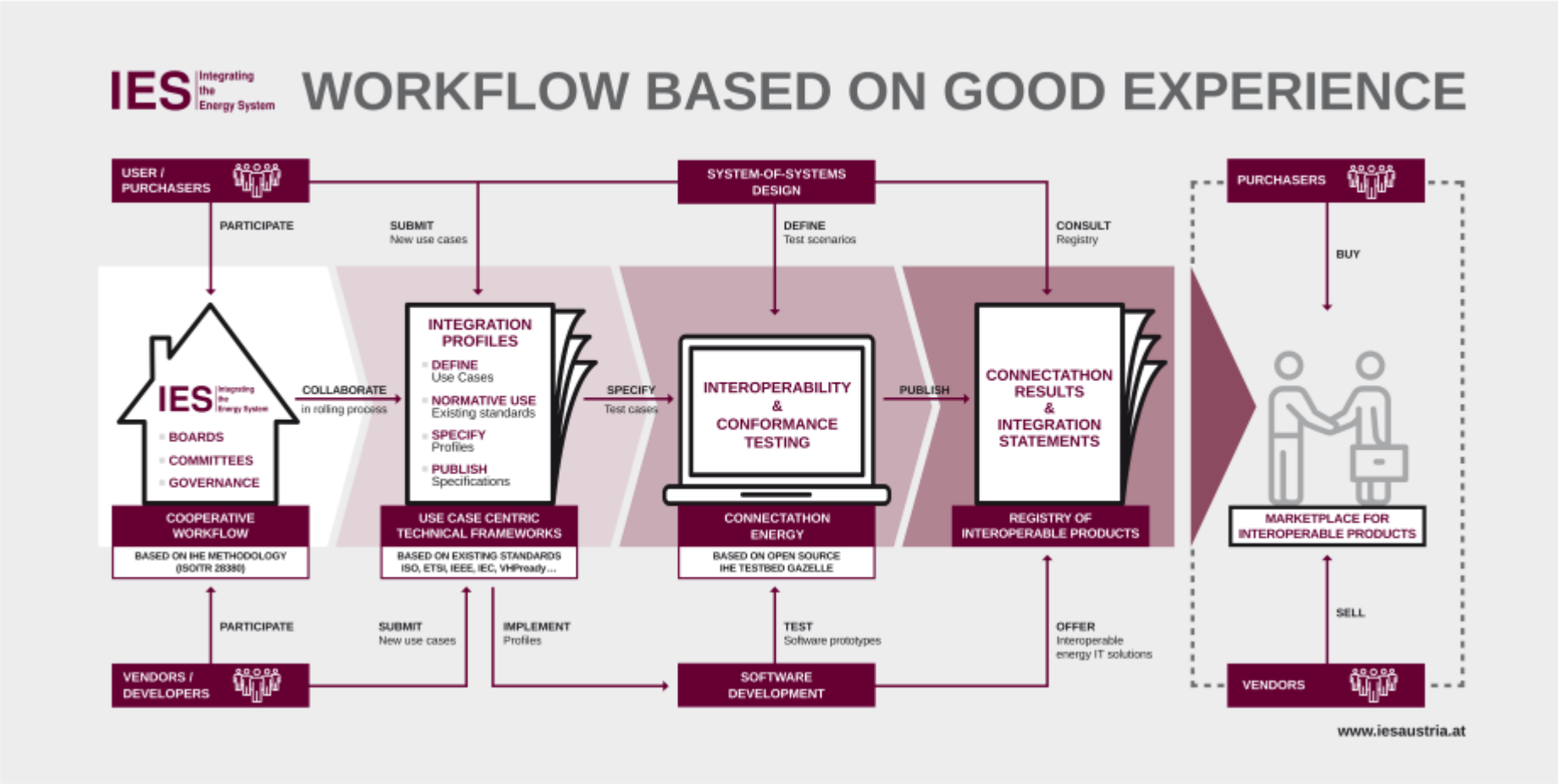
[11] K. Forsberg, H. Mooz and H. Cottermann, Visualizing Project Management – Models and Frameworks for Mastering Complex Systems, vol. 3, John Wiley & Sons Inc., 2005.

[12] E. Analytics, "An introduction to the European Interoperability Reference Architecture (EIRA) v2.1.0," Europa Analytics, 2016.

[13] M. Masi, T. Pavleska and H. Aranha, "Automating smart grid solution architecture design," in *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2018, pp. 1-6.

[14] C. Carr, "IHE international, incorporated principles of governance".

[15] "IHE profile design principles and conventions," [Online]. Available: https://wiki.ihe.net/index.php/IHE_Profile_Design_Principles_and_Conventions.



The IES process in some more detail:

- Specifying a Profile has four steps: identify the issue, agree on a solution, specify the solution, publish a new profile.
- Testing Profiles has three steps: specify test cases, prepare test tools, execute testing.
- Results has two steps: show who succeeded implementing which profiles, state which products support which profiles.
- To manage all that, experienced teams constitute dedicated committees that take on the required process management roles.

IES Integrating the Energy System

Contact:

Dr. Angela Berger, Managing Director
Technology Platform Smart Grids Austria

Homepage: www.iesaustria.at
e-mail: ies@smartgrids.at

http://www.iesaustria.at