

OPEN NETWORKING
FOUNDATION



ONF presentations to Information Modelling Workshop

ONF Common Information Model

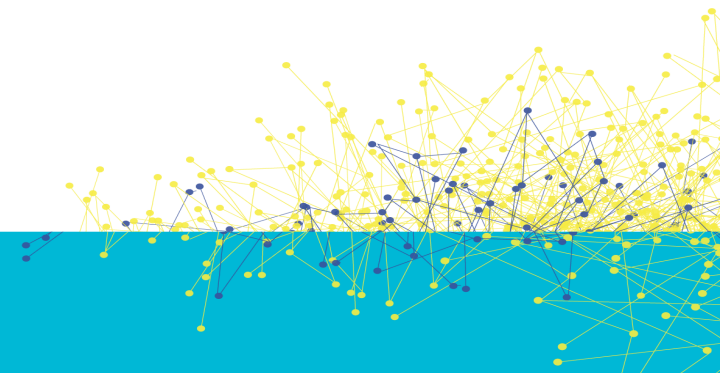
Core Information Model and Tapi update

December 2016

Contents

- Joint work update
- Recap of model
- ONF model update
- Tapi update
- Interface evolution
- Next steps for the ONF collaborative work

Note that issue identifiers form [ISSUES_SORTED_AND_CLUSTERED](#) provided where appropriate as [Louisville <ID>xx]



Joint work update

ONF/MEF/TMF (progressing but not quite there yet) [\[Louisville 11, 16\]](#)

- Legal agreement signed between ONF, MEF and TMF to enable joint working
 - This took a little longer than desired but no longer than expected.
- To participate, member companies need to sign contributor agreements
 - This is progressing gradually as member companies assess the legal agreement
- At this point traditional liaison is still being used to exchange information currently. This would be too slow but fortunately
 - The ONF work is published very openly
 - The ONF team publish a snapshot of work including all experimental work etc
 - Individuals who are members of both ONF and MEF or TMF can help explain the raw ONF documentation in MEF/TMF without violating any copyright etc
- If new features are required by MEF/TMF to be supported in ONF this still needs to go via liaison
 - TMF analysis work is still not visible to ONF... the fast pipe is rather unidirectional... joint working will speed up this process
 - ONF is expecting a liaison from TMF identifying desired refinements

ETSI NFV liaison to ONF on touch points [\[Louisville 8, 18\]](#)

- Demonstrates the critical part ONF work plays in NFV deployments
- Provides some pointers to an opportunity for convergence
- Addresses the “underlay” relationship of ONF to NFV but not the recursion [\[Louisville 10\]](#) which still needs further discussion

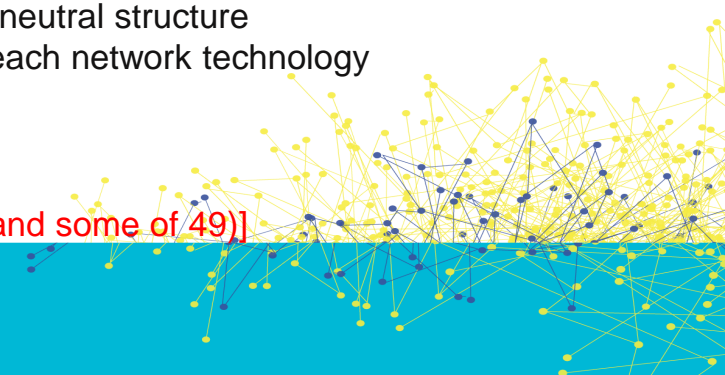
ONF/ITU-T [\[Louisville 13\]](#)

- ITU-T provides a critical focus on specific network technologies
- ONF provides the model structure of representation of networking in both SDN and traditional context (same model parts for both contexts)
- ONF technique for extension via spec model (UML) provides a simple and elegant mechanism for decoupled interrelationship between the technology specific work and the technology neutral work
- TAPI uses the spec model technique to introduce technology specific parameters into the technology neutral structure
- The ONF spec model technique allows for controlled vendor variety in implementation of support for each network technology
 - MEF has also adopted the ONF spec model technique

TAPI (Open Source)

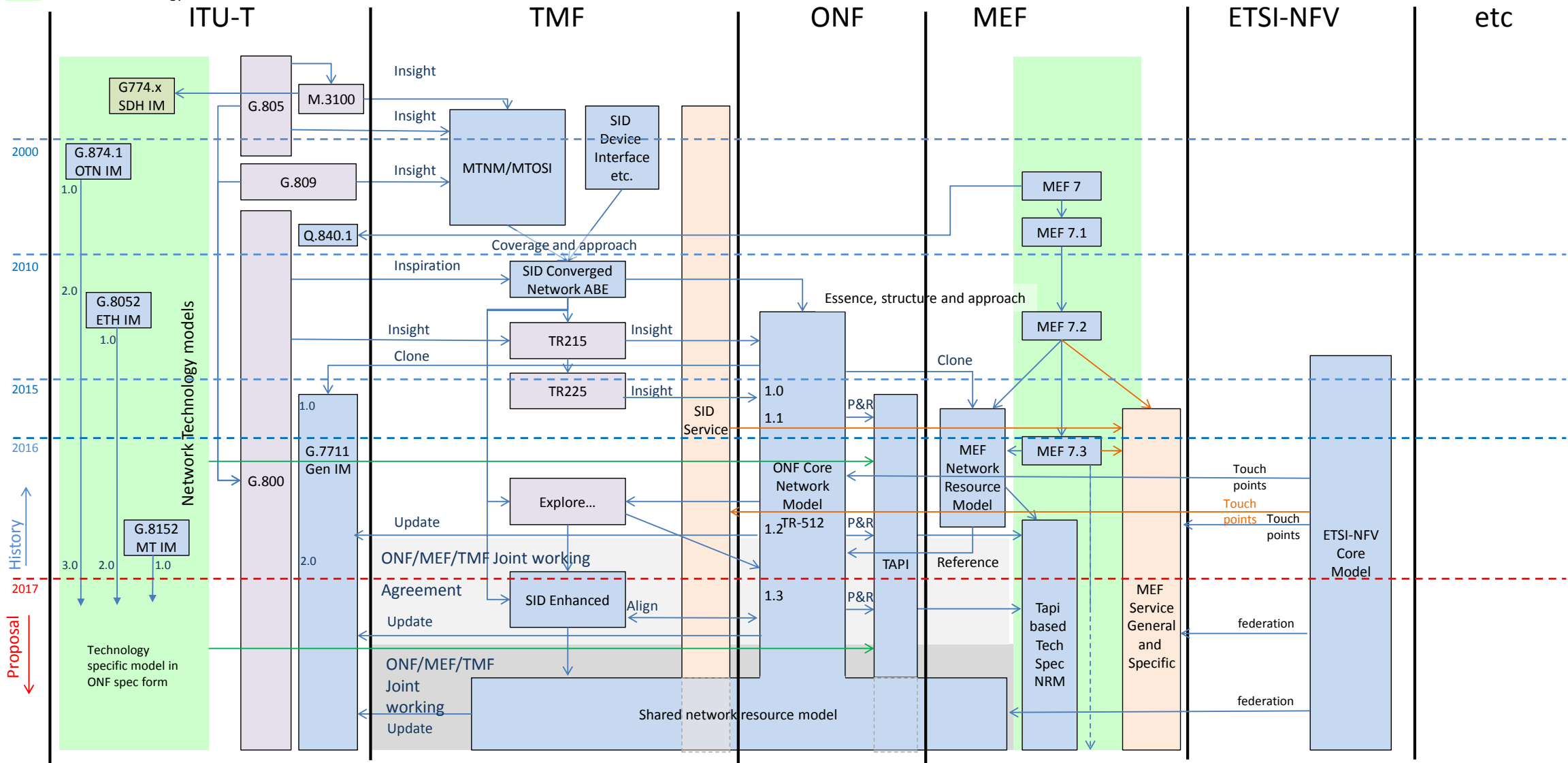
- MEF have adopted Tapi and tooling approach [\[Louisville 20, 32, 12b\]](#) for their Presto interface

Note that NMS session (Tuesday/Wednesday) covered corresponding architecture [\[Louisville 12, 40, 43 \(and some of 49\)\]](#)



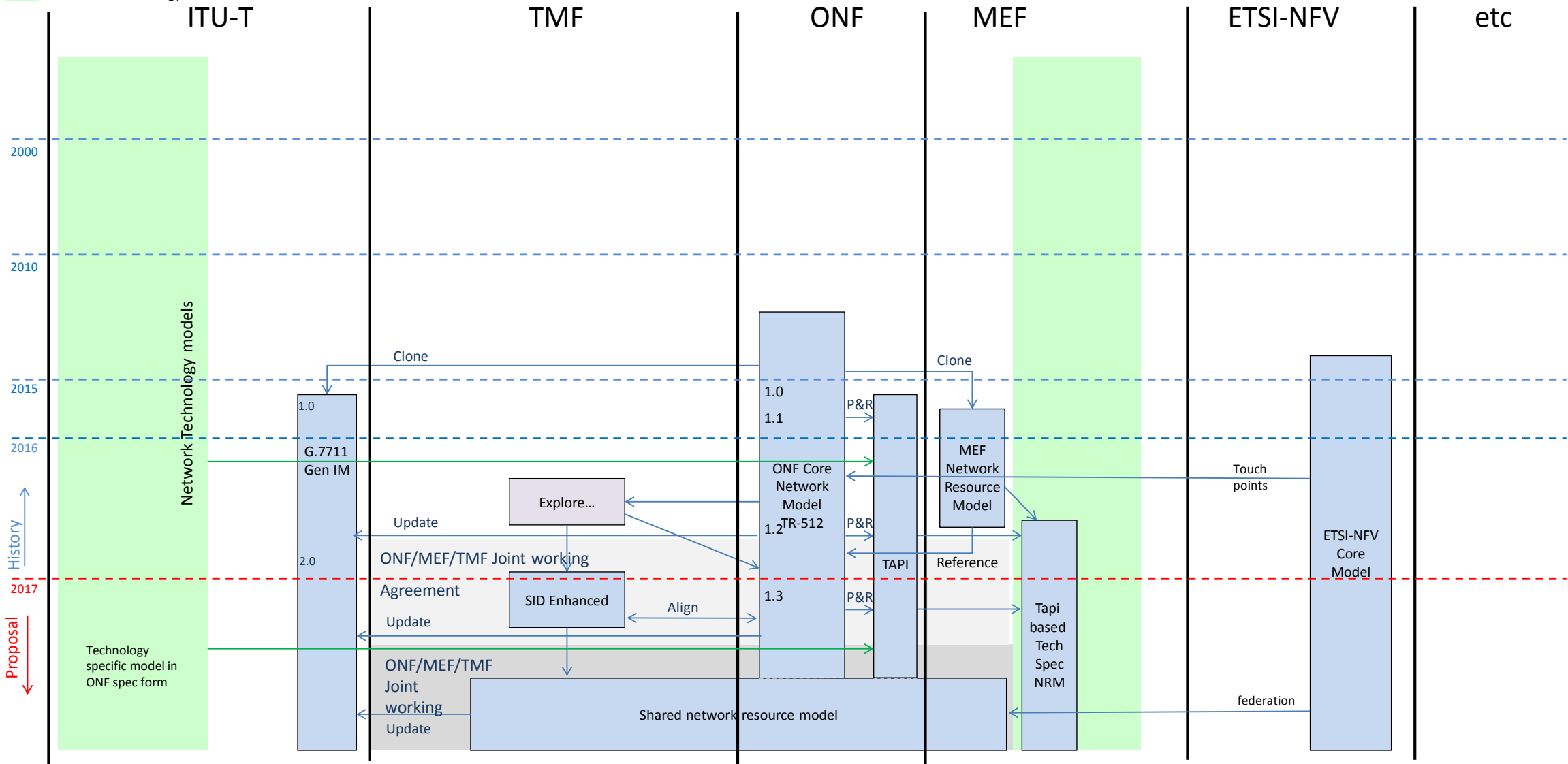
- Formal UML model
- Concepts
- Network Technology Definition

Joint work update



- Formal UML model
- Concepts
- Network Technology Definition

Joint work update – ONF focus



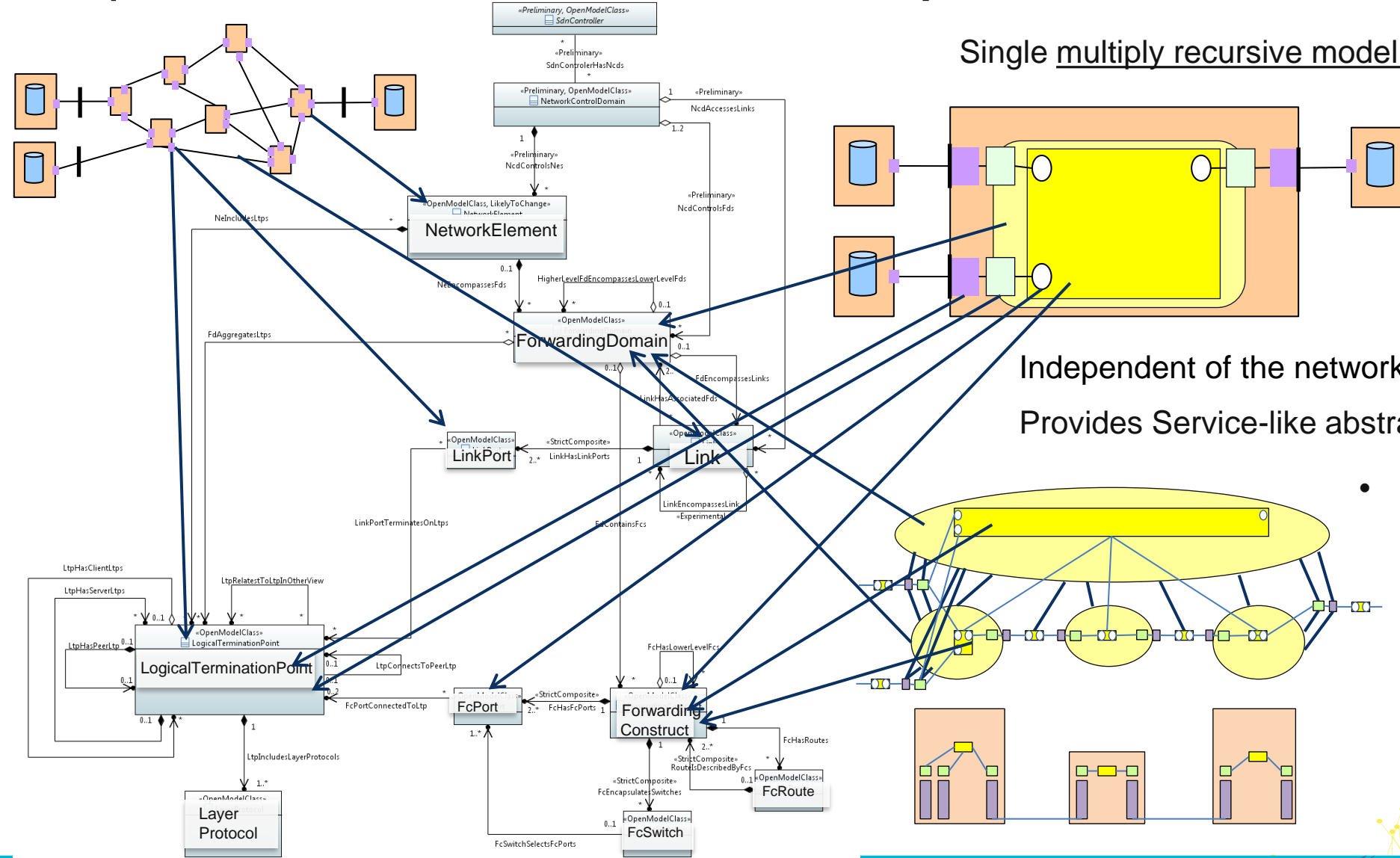
Recap... Skeleton Class Diagram of key object classes (Core Information Model Version 1.2 September, 2016 ONF TR-512)

Single multiply recursive model for representing networking

Shown in a physical context but equally represents abstract and virtual forms

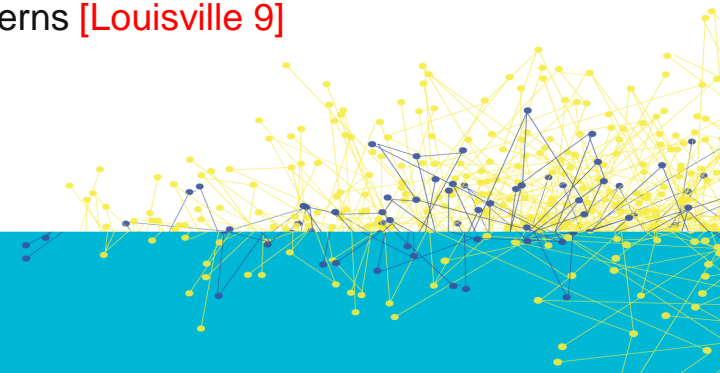
Independent of the network technology, applicable to all
Provides Service-like abstraction of the network

- Covering all key aspects:
 - Topology
 - Forwarding
 - Termination
 - Identity and naming
 - State
 - Capability specification

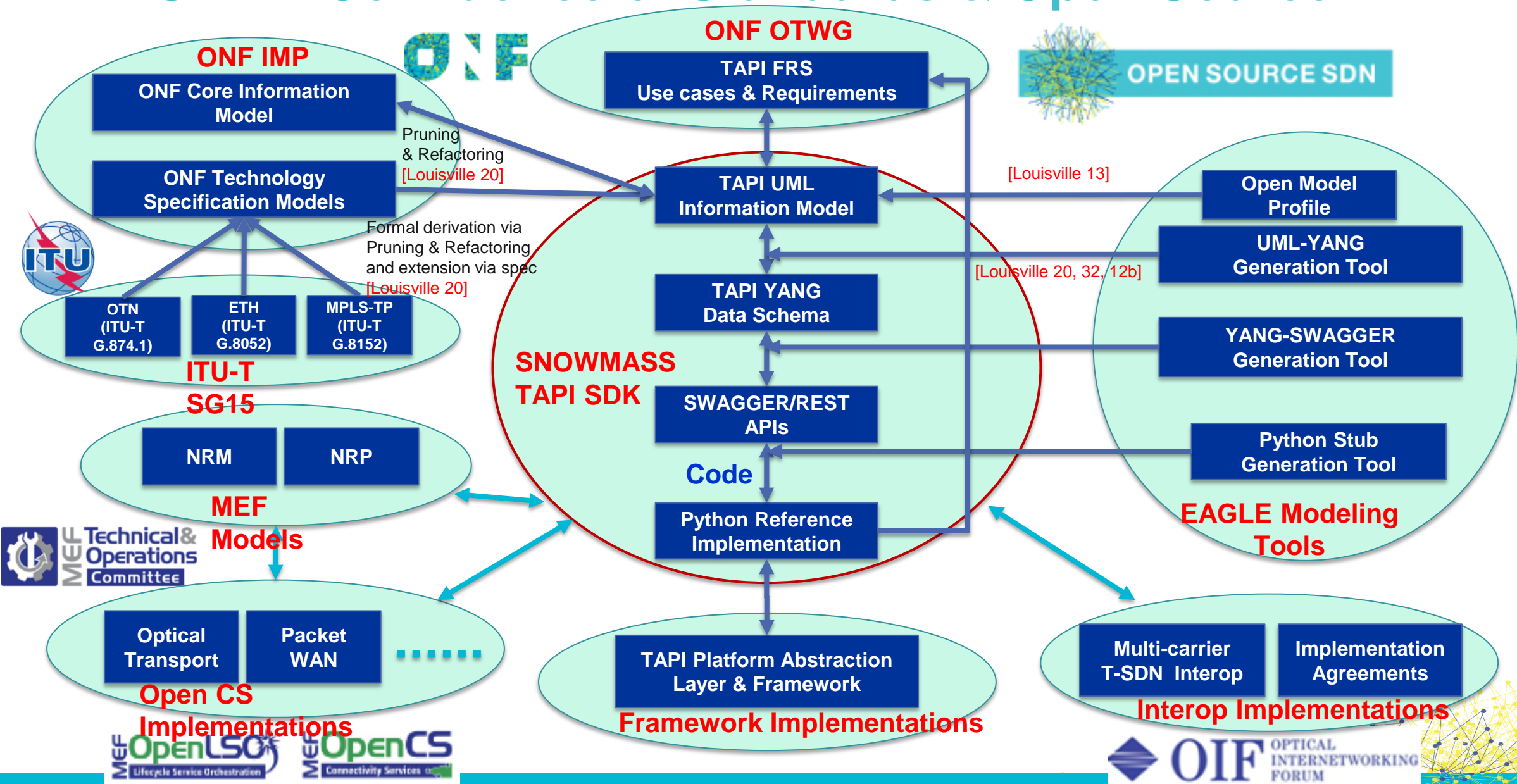


ONF model enhancements

- In version 1.2 ([Core Information Model \(CoreModel\) 1.2](#) (September 2016 | TR-512) in the [ONF technical library](#))
 - Documentation restructuring and content improvements
 - Model and documentation delivered in a single .zip file ([Core Information Model \(CoreModel\) 1.2](#) (September 2016 | TR-512)) in the [ONF technical library](#)
 - The descriptions of the model are now broken down into a number of separate documents (TR-512.1 – 11)
 - Specific document on mapping of terms provided (TR-512.9) [\[Louisville 19\]](#)
 - Enhanced documentation of specification model (TR-512.7) and introduction of the spec pattern
 - The spec model is applied in TAPI
 - Major Resilience model enhancements (TR512.5)
 - The resilience model is precise but generalized to be applicable to any protection restoration scheme in a forwarding context
 - Introduction of a Physical model focussing on Equipment (TR512.6) [\[Louisville 12a\]](#)
 - a physical thing is something that can be measured with a ruler
 - functional things are not physical
 - Even for a mechanical devices (e.g. a fan) the function is emergent and abstract (hence not physical)
- In progress for 1.3
 - Enhancements to the Physical model
 - The strengthening of the model of the relationship between physical and network (functional) models
 - Model of controller functions (as an aspect of the Management Control Continuum) [\[Louisville 12, 12a\]](#)
 - Removal of the NE and replacement with a control scope/processing construct model [\[Louisville 12a\]](#)
 - Development of a model for the Component-System pattern and other solution space patterns [\[Louisville 9\]](#)
 - Minor enhancements to support the Synchronisation/Timing network model
 - Minor enhancements to the network model support the Media Channel model
 - Validation of model support for G.8032 (Ethernet Ring Protection)
 - General enhancements to the network model



TAPI SDK – Confluence of Standards & Open Source



ONF IM team consideration on interface evolution

- Assuming a cloud oriented future, with cloudified controllers, inter-controller communication (and some intra-controller communications) will be via native cloud interface encodings
 - Cloud presents true APIs in a programming language
 - Mapping to the communication infrastructure is provided by the cloud infrastructure
- Interface interaction will benefit from normalization of interaction patterns and a messaging grammar that unifies the variety of interaction complexities (CRUD, Intent etc) in a single sophisticated structure
 - The key is modelling the message content structure... work is underway in ONF on such a model
 - The structure definition will enable folding away of capability that is unnecessary for any specific interaction
- Essential to any interaction is the provision of information about the desired outcome in terms of constraints and potentially in the context of some expected initial system state
- The content of any message will differ per interaction
 - The key to content opportunity for the viewpoint shared between the parties at the interface is determined by the properties of the attributes in that viewpoint (read only, read-write etc)
 - The information about any particular context or outcome can be a flat structure of things and their attributes that is as simple as possible for the specific interaction
 - The structure can vary from interaction to interaction
- As understood from various activities over the past decade or so the concept of an NE as a thing is broken
 - The ONF work activity mentioned earlier “Removal of the NE” will address this using insights gained in those various activities
 - The NE dematerialized into at least a geographical physical aspect, controlled functions aspect and controlling functions aspect
 - The part of the NE that the controller communicates with is the controlling functions aspect, this is just another controller
 - The controller model is under development and this will be applied
- Emergence of white box NE points to cloudification of the controller functions in the NE
 - In future interface to controller directly attached to the network functionality will via cloud native interfaces as for any other controller
- In the above environment interfaces generated from canonical models via tooling will be the norm



Next steps for the ONF Multi-SDO collaborative work

To be progressed as a focus:

- TOSCA [Louisville 41, 44]
 - Applying the ONF IM and Tapi in a TOSCA context
 - Discussing the ONF interface vision
- IETF
 - Analysis of TEAS Topology and mapping it from ONF core using pruning and refactoring process [Louisville 20, 32, 12b]
 - Consider comparing and contrasting the microwave model work on ONF and IETF
- TMF interface work and Tapi
 - Application of the ONF Spec model in a TMF context
 - Discussing the ONF interface vision

Still a major challenge:

- Open source \leftrightarrow standards interaction [Louisville 26, 49]
 - Recent ONF – OnLabs consolidation will be a catalyst for clarity and progress in this aspect
- Legal structure of joint working arrangements [Louisville 31, 40a]
 - Whilst the ONF-MEF-TMF agreement is now in place this was very challenging
- Determining a definitive list of Yang best practices
 - Best practices not easily identifiable
 - Some work in ODL but not clear whether this is the definitive statement
 - Some relevant material in RFC7223
 - Note that Tapi does not leverage RFC7223 at this point
 - Relevant material in RFC6087 <https://tools.ietf.org/html/draft-ietf-netmod-rfc6087bis-09>
 - Work with IETF etc



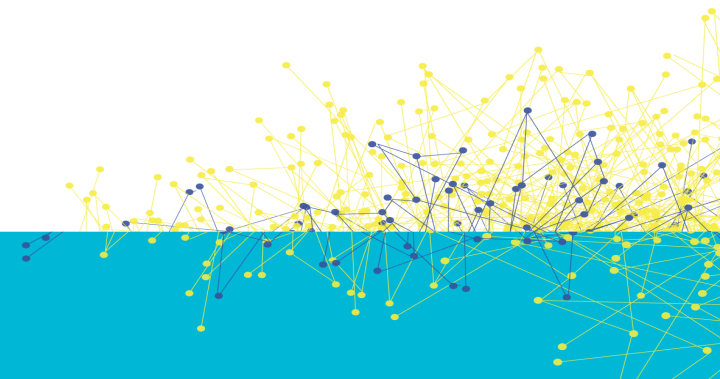


THANK YOU



Background material

Key material presented in the last session in January



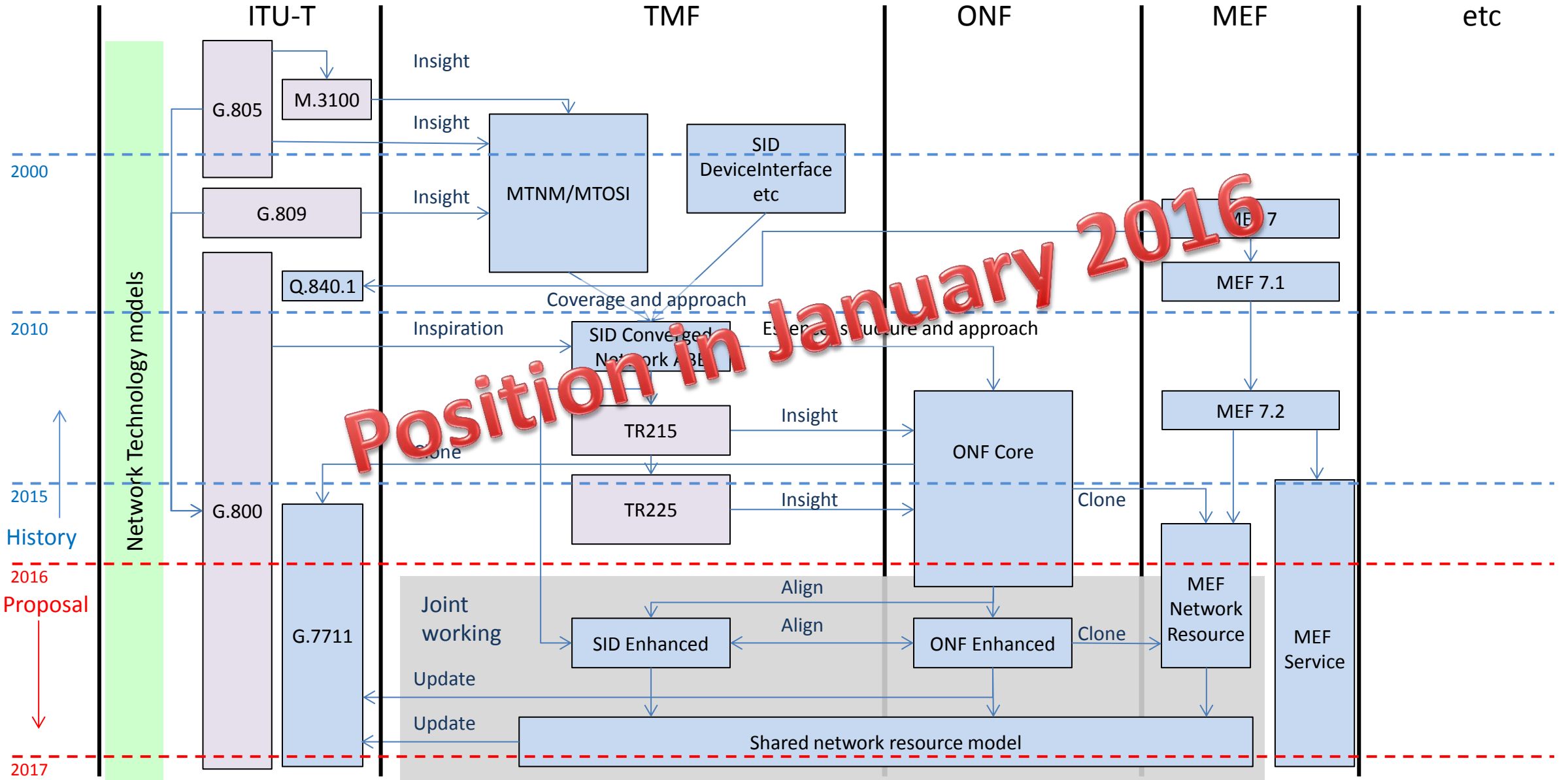
Rationale and approach

- Information modelling is the process of identifying, defining, labelling and recording, from some viewpoint
 - The key concepts/things in the problem space
 - The interrelationships between those concepts/things
- The information model provides
 - Terminology definition and terminology normalization
 - Enabling constraints, drawn from industry insight, to prevent errors and unnecessary excursions
 - The basis for all interface views of the domain
- The process for development is “Agile Architecture” applied to the on-going evolution of the representation of the problem space
 - Building on existing work of the industry pulling in experts in these areas
 - Development by contribution and collaborative working refining and improving each aspect of the solution
 - Developing industry-wide guidelines for modelling
 - Use of open source tooling (Papyrus, Gendoc, Git etc.) and extending that tooling as required
 - Incremental publication of a snapshot of all work including experimental material
 - Gradually increase work in open modelling activities
 - Driving/guiding interface code development
- The interface model views are constructed using the process of pruning/refactoring to transform the core model for external presentation
 - The interface encoded form is increasingly generated by tooling



Model Evolution History and Proposal

- Formal UML model
- Concepts
- Network Technology Definition



Key sources and artefacts of the ONF-CIM

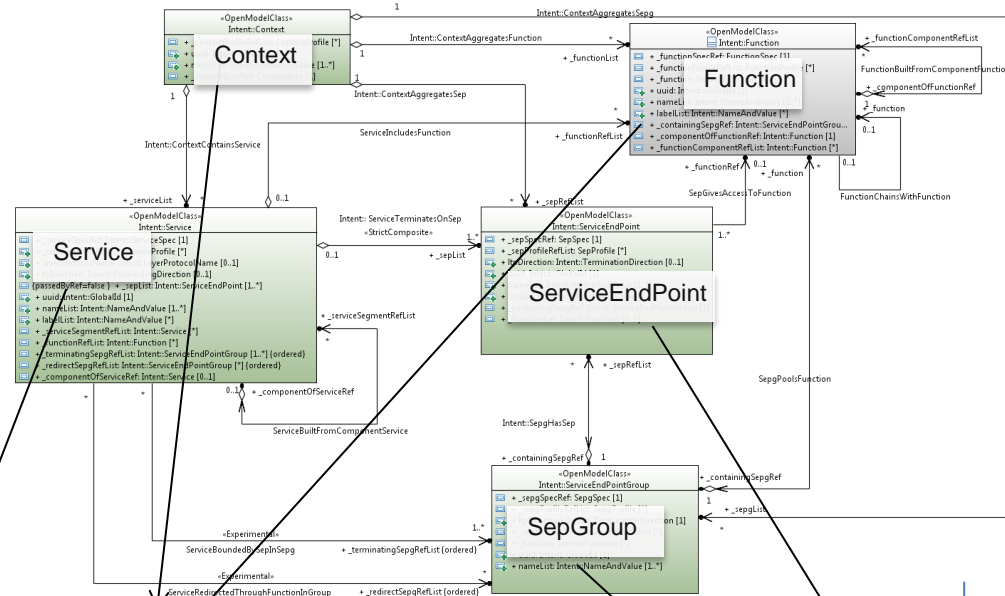
- The ONF-CIM is based on
 - G.800/G.805 network architecture
 - Subnetwork, Link, Adaptation/termination
 - Layer networks, partitioning, recursion
 - Insights from TMF TR215/225+
 - Hypergraph, component-system, ForwardingConstruct
- Key object classes (illustrated on next slide)
 - ForwardingDomain (FD) – derived from G.800 subnetwork
 - ForwardingConstruct (FC) – derived from G.800 forwarding relationship
 - Link – derived from G.800 Link
 - LogicalTerminationPoint (LTP) – derived from an assembly of G.800 termination and adaptation
- Common core class definition
 - Extensions (via conditional packages) to provide technology specific classes



Transforming the Core Model into the “Boulder” Intent Model

Prune And Refactored Core

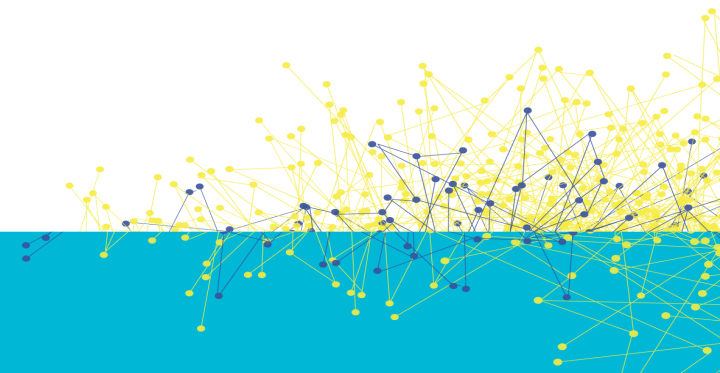
- Service is FC
- Context is FD
- SepGroup is LTP
- Sep is LTP



- Currently playing into OpenStack/ODL via the Boulder OpenSource project
- Intent work focuses on constraint based outcome oriented interactions
- The intent model is derived from the CIM by pruning and refactoring
 - Includes mapping object names

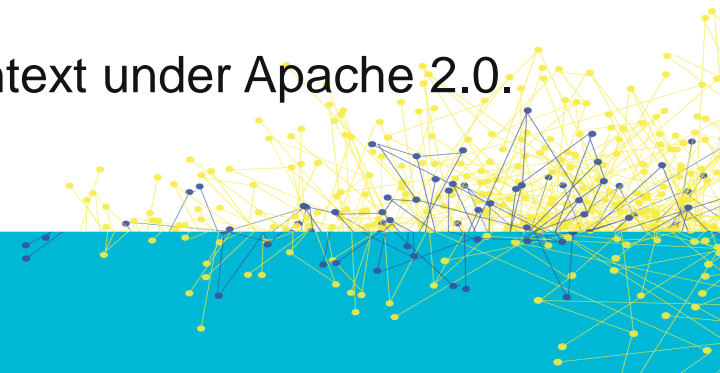
Key Insights used to design the ONF-CIM

1. Results from many years of theory and practical insight
2. Based on key patterns such as “component-system” and “encapsulation”
3. Provides a canonical model of “enabling constraints”
4. Modularized with light coupling to enable parallel working and independent evolution of parts
 - Use associations between modules to maintain flexibility
 - Extended by composition (rather than sub-classing)
5. Does not include operations in the canonical model but instead provides all necessary property constraints to govern operation generation at a later stage on the process



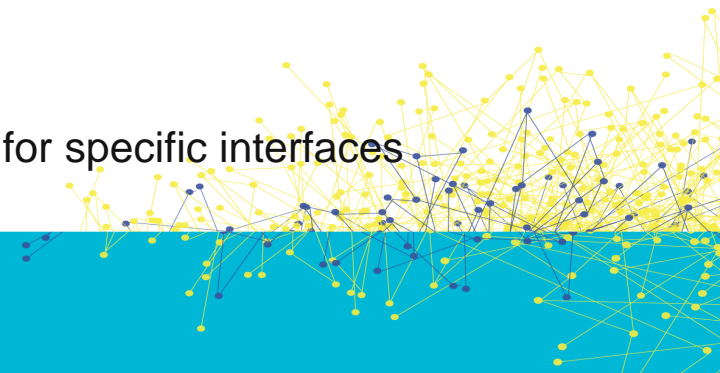
Key Advantages of using the ONF-CIM

1. Widely applicable model that uses an optimised compact assembly of classes
2. Focuses on the domain of Forwarding and Termination (in the Core Network Module)
3. Network technology neutral
 - Supports technology specific extensions
4. Extended by composition (rather than sub-classing)
5. Provides model for identifiers, names and states (in the Core Foundation Module)
6. Published as a snapshot of all current work including approved, preliminary and experimental parts
 - Maturity shown using the lifecycle stereotypes of the Open Model Profile
7. Modularized with light coupling to enable parallel working and independent evolution of parts
8. Oriented towards interface generation
9. Allows controlled adjustments via pruning/refactoring in interface development
10. Uses tooling to generate interface schema from the model
11. Developed using open source tooling with robust guidelines
12. Progressing parts to publication and evolution in an open source context under Apache 2.0.
13. Already used by several SDOs (see next)



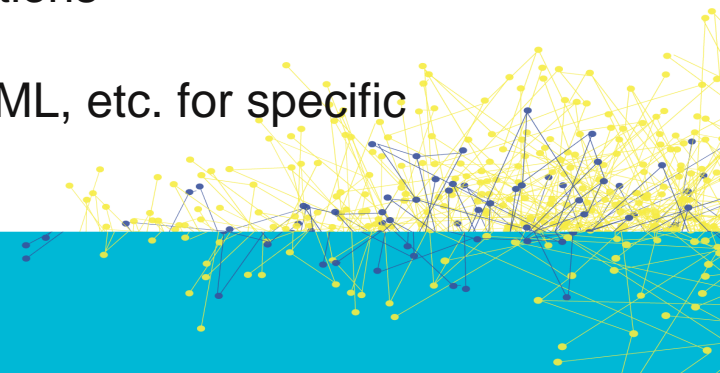
ONF CIM Core Model already adopted by many projects

- External to ONF
 - ITU-T: Transport resource and networking models
 - Published as G.7711 (ex. G.gim)
 - Extension & normalization of technology-specific fragments: G.874.1 (OTN), G.8052 (ETH), G.8152 (MPLS-TP)
 - MEF: Cloned to form MEF Network Resource Model
 - BBF: PoC exercise – simple firewall object for the broadband fragment
 - OIF: PoC using Tapi which is derived from the Core Model
- All IM projects in ONF (including ONF Sponsored open source activities)
 - Extension for technology/application-specific fragments of the ONF-CIM
 - OTN, Carrier Ethernet, and MPLS-TP, Wireless Transport
 - Interface-views for specific purposes through pruning & refactoring
 - Transport API: e.g., topology/connection/path-computation/virtual-network services
 - North Bound Interface (NBI): e.g., Topology/Inventory/Service-Flow services
 - Intent: Client-focussed outcome-oriented constraint-based interactions
 - Interface encoding
 - Facilitate tooling-enabled mapping to OpenFlow, YANG, JSON, XML, etc. for specific interfaces



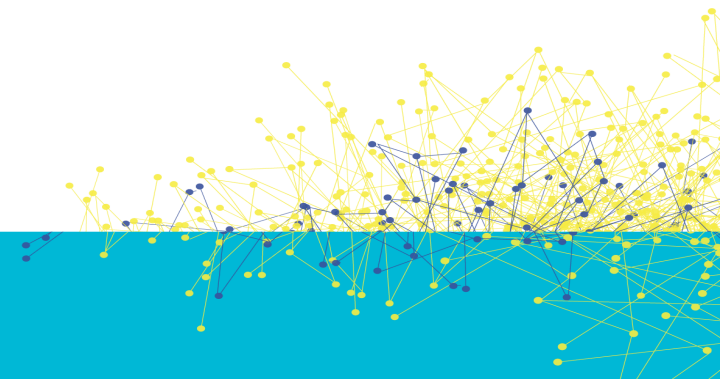
Update on ONF CIM Core Model adoption

- External to ONF
 - ITU-T: Transport resource and networking models
 - Published as G.7711 (ex. G.gim)
 - Extension & normalization of technology-specific fragments: G.874.1 (OTN), G.8052 (ETH), G.8152 (MPLS-TP)
 - MEF: Cloned to form MEF Network Resource Model
 - OIF: PoC using Tapi which is derived from the Core Model
- Work in ONF
 - Extension for technology/application-specific fragments of the ONF-CIM
 - OTN, Carrier Ethernet, and MPLS-TP, Wireless Transport
 - Interface-views for specific purposes through pruning & refactoring
 - Transport API: e.g., topology/connection/path-computation/virtual-network services
 - North Bound Interface (NBI): e.g., Topology/Inventory/Service-Flow services
 - Intent: Client-focussed outcome-oriented constraint-based interactions
 - Interface encoding
 - Facilitate tooling-enabled mapping to OpenFlow, YANG, JSON, XML, etc. for specific interfaces



Background material

Information Modelling rationale and approach



Information modelling

- We all do at least informal information modelling in our day to day work
- Information modelling is simply the process of identifying, defining, consistently labelling and recording, from some viewpoint:
 - The key concepts/things in the problem space
 - The interrelationships between those key concepts/things
- When we analyse a problem we develop/use terminology and develop structure
 - But there is always a challenge of colliding terms and apparently contradictory structure
- A key purpose of the ONF work is to provide “enabling constraints” and an appropriate degree of normalisation
 - Enabling constraints prevent errors and unnecessary excursions
 - Normalisation remove unnecessary variety, emphasises only inherent complexity, allows attention to be focussed on areas requiring exploration and improves interoperability
 - The key is the right degree of normalisation so as to not stifle innovation but instead to enable it
- Formal Information Modelling is a technique that supports this purpose for things to be discussed in the solution especially over interfaces



Key considerations (1/5)

To what degree should we normalize each area/domain/concept?

- ONF focus has been the domain of networking and forwarding which is a mature domain
 - Within that domain there is an opportunity to have deep normalization of concepts based upon ITU-T G.80x and the related TM Forum Converged Network model
 - This model is within the Core Information Model
 - There is very deep pattern normalization but still some terminology challenges
- In general the degree of normalization depends upon the maturity of the domain (see next slide)

Which views are important to focus on?

- ONF focus has been on several views
 - An interface oriented information architecture (G.805/800 work has inspired a more pure information architecture in TM Forum – this was liaised to ONF) that is network technology agnostic that provides
 - The essential model of the problem space
 - The rationale for various interface views
 - A bridge between those interface views
 - A generalized model of naming and identity
 - Specific technology additions via a compositional extension approach
 - A number of interface views for specific purposes where those views are derived from the Core Information Model
- In general the focus depends upon purpose but for standardisation initiatives where interfaces and reference points are the key interoperability output the above focus would appear to apply

Initial attempt
Raw Reality Models

Normalization

Subsequent attempts
Crude Initial Classification
Models

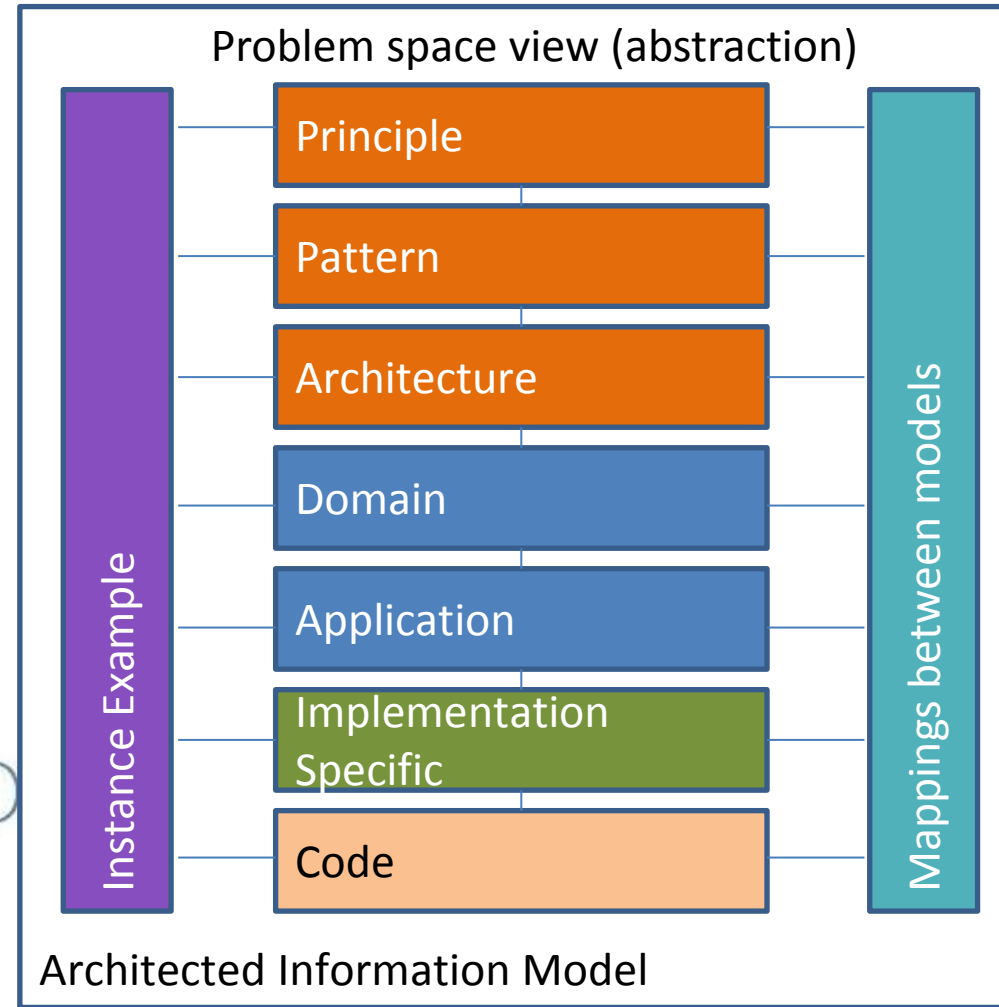
Mature Classification Model

Increasing maturity with time

Unnecessary variety decreases over time

Normalisation
opportunity

Resolves to only the inherent
complexity



Key considerations (2/5)

What approach should we use to develop & to apply the Information Model?

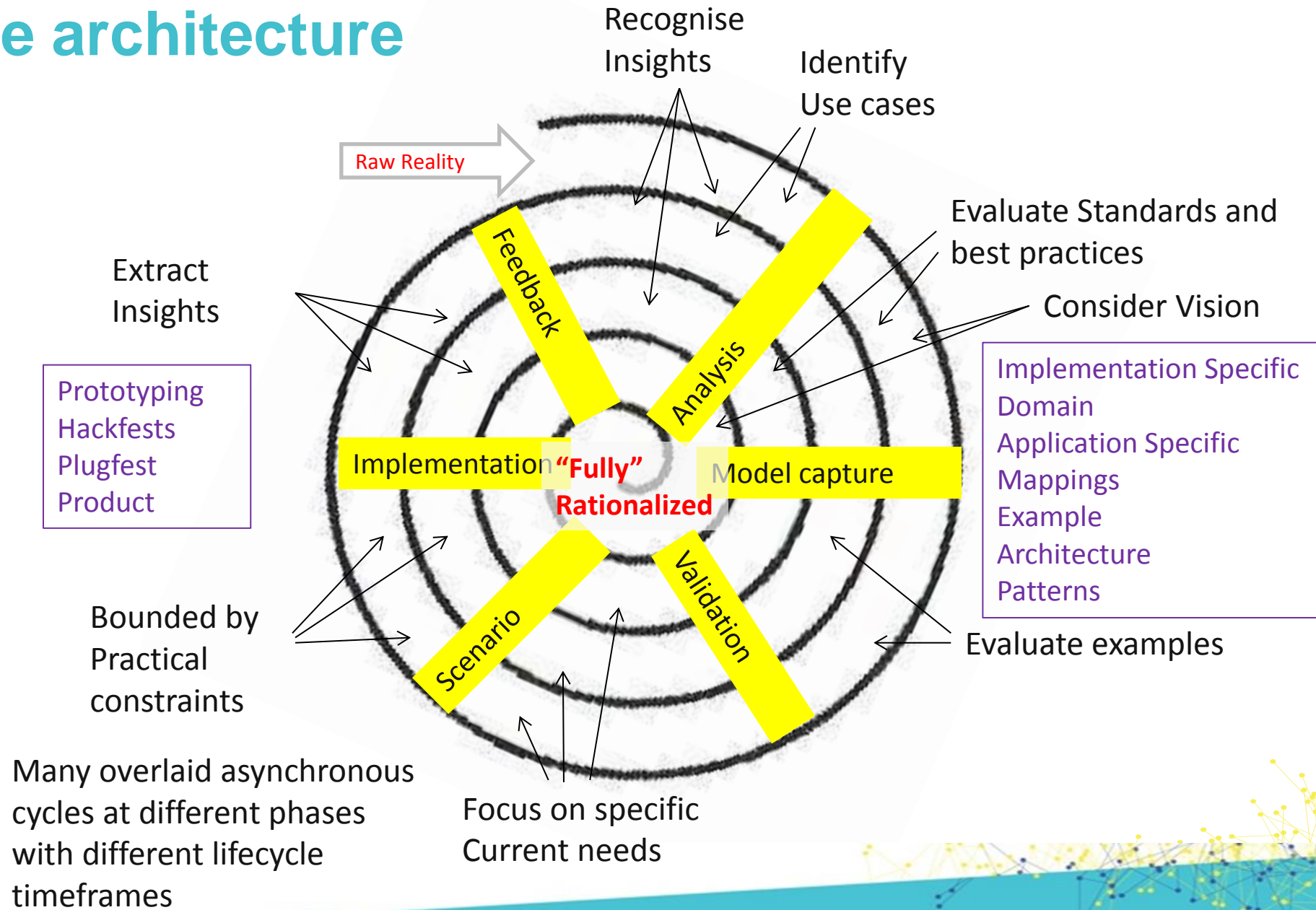
- ONF has used an agile architecture approach
 - The model development takes on insights from the industry in the context of the SDN vision
 - The deliverable includes both formalized and agreed model as well as preliminary and experimental parts
 - The deliveries occur every 3 to 6 months
 - The interface models are being developed by appropriate pruning and refactoring of the core model
 - The interfaces are being validated via practical application with the results being fed back into the model evolution process
- In general an agile approach is recommended with the delivery including preliminary work to help decision making in forward looking development enabling intercepts to be developed and that deliver enhancements rapidly as they emerge (see next slide)

What are the inputs to the modelling process?

- ONF has drawn from existing work in the industry, from SDN definitions and from SDN vision
 - Drawing insights from ITU-T, TM Forum, OIF, MEF etc and collaborating in that work
 - Applying the recursive control architecture and the need for abstraction of the network to client views
 - Supporting Openflow needs
 - Validating with interface use cases
- In general successful solutions emerge from development approaches that take advantage of a conjunction of theory and practical deployment experience.
 - Clearly the model should be positioned to match the vision, architecture and interface needs



Agile architecture



Key considerations (3/5)

How much should we formalize the activity of Information Modelling?

- ONF has chosen to strongly formalize information models for both the essential problem and the interfaces using UML
 - It was concluded that complex meshy problems benefit from a strong model and that UML provides an appropriate environment
 - To formalize the model it is necessary to use suitable tooling and in ONF Papyrus (free open source Eclipse platform tool) has been chosen. Incidentally this is also used in ITU-T and MEF
 - Formalization means consistent application of UML and to ensure this guidelines have been developed
- In general it is recommended that complex areas (and most actually turn out to be) should be modelled with UML. The ONF guidelines provide a good basis for Information Modelling in any complex problem space

Who develops the Information Model, when and how?

- ONF has chosen to set up an Information Modelling team to own the Core model on an on-going basis and to coordinate other modelling activities. There are also separate teams working on the interface models and on network technology enhancements
 - Interface development uses a pruning and refactoring approach to develop interface that allows loose coupling between the Core team and the Interface development teams
 - Modelling can occur in parallel in the various teams as each has been allocated a suitable branch in Git
 - The model ownership has been clearly demarked
- In general it is recommended that an ownership approach that partitions the work to minimize dependencies is developed and for work on solutions similar to that of ONF an approach similar to that chosen in ONF is recommended



Key considerations (4/5)

Who uses the Information Model, when is it used and how?

- In ONF the:
 - Core model is used when advancing the understanding of the problem space.
 - Development is done incrementally drawing from expertise and industry insight as well as from feedback from the interface work
 - Core model is used when developing interface definitions
 - As noted before by pruning and refactoring
 - Interface models are used when developing interface schemas (e.g. JSON or Yang)
 - Currently manual but to specific guidelines and aiming at automation
- In general it is important that the role of each aspect of the model and the relationships between each aspect is explained well and understood. It is expected that a similar approach would be valid for any other body developing interfaces

How should we represent the formalized Information Model?

- The ONF model is provided in both a Papyrus UML model form and a document form.
 - It is supported by pictorial representations in a domain specific language
- In general it is suspected that a similar approach of tooled model, documentation and supporting material would be applicable



Key considerations (5/5)

How do we deal with variety, extensibility and change?

- The ONF model uses several approaches:
 - The lifecycle of extensions/changes are indicated using the Lifecycle stereotypes
 - Changes are coordinated by the Information Model team with the aim of maintaining stability and fostering convergence
 - Variety is allowed between views where it is driven by external need but each view needs to relate to the common core model
 - Network technology specific attributes are added in conditional packages by the network technology experts
 - There is a recognition that there is unnecessary variety between network technologies and there is work underway to help remove that variety
 - Coordination using git
 - It should be noted that the model is an on-going development
- In general it is recommended that a similar approach with lifecycle stereotypes, model governance and partitioned ownership is followed

How many Information Models?

- ONF essentially have one model made of many model modules, some of which are stand-alone and could be considered as models in their own right
 - Hence it is both one model and many models... or all are fragments of one model
- In general a model is in many fragments. Where there are fragments the key is to coordinate/collaborate
 - Apply consistent terminology defined in terms of structure of a model
 - Collect domain experts to cover each area
 - Be pragmatically holistic

Benefits of the core model

- Help to reduce unnecessary variety in the industry
 - A major and essentially removable cost is that of integration of solution elements that use unnecessarily different ways to represent what are essentially common concepts.
- Assist in sharing of the fundamental concepts to enable linkage between management and control environments
 - Because of the linkage to traditional models the Core Model Fragment will enable relatively easy integration into existing management/control environments
- Assist driving mappings between OF and Service level abstractions (with the Core Model as an intermediary) improving coherence and enabling the necessary bridge between OF and Application models
 - The key concepts in the Core Model Fragment can be related through defined specification classes in the Core Model Fragment to the TTPs (as principles behind the TTPs are similar to those behind the core model)
- Provide a formal base for the work of the NBI /Transport API teams improving ONF internal coherence

Through this model assist in the rationalization across the industry of the representation of networks for the purpose of management-control

