ECOMP (Enhanced Control, Orchestration, Management & Policy) Architecture White Paper

AT&T Inc.

Abstract: AT&T's Domain 2.0 (D2) program is focused on leveraging cloud technologies (the AT&T Integrated Cloud – AIC) and network virtualization to offer services while reducing Capital and Operations expenditures and achieving significant levels of operational automation. The ECOMP Software Platform delivers product/service independent capabilities for the design, creation and lifecycle management of the D2 environment for carrier-scale, real-time workloads. It consists of eight (8) software subsystems covering two major architectural frameworks: a design time environment to design, define and program the platform, and an execution time environment to execute the logic programmed in the design phase utilizing closed-loop, policy-driven automation.

ECOMP is critical in achieving AT&T's D2 imperatives to increase the value of our network to customers by rapidly on-boarding new services (created by AT&T or 3rd parties), enabling the creation of a new ecosystem of cloud consumer and enterprise services, reducing Capital and Operational Expenditures, and providing Operations efficiencies. It delivers enhanced customer experience by allowing them in near real-time to reconfigure their network, services, and capacity. While ECOMP does not directly support legacy physical elements, it works with traditional OSS's to provide a seamless customer experience across both virtual and physical elements.

ECOMP enables network agility, elasticity, and improves Time-to-Market/Revenue/Scale via the AT&T Service Design and Creation (ASDC) visual modeling and design. ECOMP provides a Policy-driven operational management framework for security, performance and reliability/resiliency utilizing a metadata-driven repeating design pattern at each layer in the architecture. It reduces Capital Expenditure through a closed loop automation approach that provides dynamic capacity and consistent failure management when and where it is needed. ECOMP facilitates Operations efficiency though the real-time automation of service/network/cloud delivery and lifecycle management provided by the Operational Management Framework and application components.

It provides high utilization of network resources by combining dynamic, policy-enforced functions for component and workload shaping, placement, execution, and administration. These functions are built into the AT&T ECOMP Platform, and utilize the AIC; when combined, these provide a unique level of operational and administrative capabilities for workloads that run natively within the ecosystem, and will extend many of these capabilities into 3rd party cloud ecosystems as interoperability standards evolve.

Diverse workloads are at the heart of the capabilities enabled by the ECOMP Platform. The service design and creation capabilities and policy recipes eliminate many of the manual and long running processes performed via traditional OSS's (e.g., break-fix largely moves to plan and build function). The ECOMP platform provides external applications (OSS/BSS, customer apps, and 3rd party integration) with a secured, RESTful API access control to ECOMP services, events, and data via AT&T gateways. In the near future, ECOMP will be available in the AT&T D2 Incubation & Certification Environment (ICE) making ECOMP APIs available to allow vendors, cloud providers, and other 3rd parties to develop solutions using ECOMP and AIC reference architecture (current and future-looking).

1. Introduction

This whitepaper is intended to give cloud and telecommunication providers, solution vendors, and other interested 3rd parties an overall context for how the Enhanced Control, Orchestration, Management and Policy (ECOMP) platform enables the AT&T Domain 2.0 (D2) initiative and operates within the AT&T Integrated Cloud (AIC) infrastructure. In order to more completely understand what drove the definition of the ECOMP Architecture and the functions ascribed to it, it is helpful to consider the

initial drivers of AT&T's D2 and network function virtualization efforts.

When the D2 effort began, cloud technology was being adopted, focusing primarily on Information Technology (IT) or corporate applications. The cloud technology provided the ability to manage diverse workloads of applications dynamically. Included among the cloud capabilities were:

- Real-time instantiation of Virtual Machines (VMs) on commercial hardware where appropriate
- Dynamic assignment of applications and workloads to VMs
- Dynamic movement of applications and dependent functions to different VMs on servers within and across data centers in different geographies (within the limits of physical access tie-down constraints)
- Dynamic control of resources made available to applications (CPU, memory, storage)

At the same time, efforts were underway to virtualize network functions that had been realized in purpose-built appliances: specialized hardware and software (e.g., routers, firewalls, switches, etc.). Network function virtualization (NFV) was focused on transforming network appliances into software applications.

AT&T's D2 strategy is grounded in the confluence of network function virtualization, Software Defined Network (SDN), and cloud technology. With virtual network functions running as cloud applications, D2 takes advantage of the dynamic capabilities of cloud cited above in defining, instantiating and managing network infrastructures and services. This strategy shaped the definition of ECOMP, its architecture and the capabilities/functions it provides. This strategy also shaped the AT&T Integrated Cloud (AIC) infrastructure that converges multiple clouds into one corporate smart cloud capable of interoperating dynamically with ECOMP-controlled virtual functions and 3rd party clouds.

In D2 the dynamic cloud capabilities are applied to applications - i.e., virtual network functions (vNFs) - thus applying the benefits of cloud to virtual network elements. For example, vNFs, such as routers, switches, firewalls, can be "spun up" on commodity hardware, moved from one data center to another center dynamically (within the limits of physical access tie-down constraints) and resources such as CPU, memory and storage can be dynamically controlled.

The ECOMP architecture and Operational Management Framework (OMF) were defined to address the business/strategic objectives of D2 as well as new technical challenges of managing a highly dynamic environment of virtual network functions and services, i.e., a software ecosystem where functions such as network and service provisioning, service instantiation, and network element deployment all occur dynamically in real-time.

ECOMP enables the rapid on-boarding of new services (created by AT&T or 3rd parties) desired by our customers and the reduction of OpEx and CapEx through its metadata driven service design and creation platform and its real-time operational management framework – a framework that provides real-time, policy driven automation of management functions. The metadata driven service design and creation capabilities enable services to be defined with minimal IT development required thus contributing to reductions in CapEx. The real-time OMF provides significant automation of network management functions enabling the detection and correction of problems in an automated fashion contributing to reductions in OpEx.

One of the challenges for service providers in a traditional telecommunications environment is the fact that unique and proprietary interfaces are required for many network management and element management systems (EMS), leading to significant integration, startup, and operational costs. Standards evolve slowly in this space.

As AT&T transitions to a SDN/NFV cloud based environment, we plan to continue contributing and leveraging the open source community to facilitate agile and iterative standards that incorporate incremental improvements. In the Domain 2.0 ECOMP ecosystem, we look to be able to rapidly onboard vendor VNFs with standard processes, and operate these resources via vendor-agnostic controllers and standard management, security, and application interfaces. Configuration and management is model driven using Yang and will utilize standards as they become available. To this end, AT&T supports open cloud standards (e.g., OpenStack, TOSCA, etc.) and engages in many Cloud and Network Virtualization industry initiatives (e.g., NetConf, Yang, OPNFV, etc.). As further

standardization occurs, AT&T will incorporate them into ECOMP as appropriate.

AT&T's objective is to virtualize and operate over 75% of target network workloads within AT&T's Domain 2.0 Architecture by 2020. Our goal for the first year is to deploy initial capabilities to validate the architecture. This early investment establishes the metadata driven foundation that enables rapid onboarding of new resources, capabilities, and services without requiring long development cycles.

2. ECOMP Platform

The ECOMP Platform enables product/service independent capabilities for design, creation and lifecycle management. There are many requirements that must be met by ECOMP to support the D2/ECOMP vision. Of those many requirements, some are key in supporting the following foundational principles:

- The architecture will be metadata-driven and policy-driven to ensure flexible ways in which capabilities are used and delivered
- The architecture shall enable sourcing best-in-class components
- Common capabilities are 'developed' once and 'used' many times
- Core capabilities shall support many AT&T Services
- The architecture shall support elastic scaling as needs grow or shrink

These capabilities are provided using two major architectural frameworks: (1) a design time framework to design, define and program the platform (uniform onboarding), and (2) a runtime execution framework to execute the logic programmed in the design time framework (uniform delivery and lifecycle management). Figure 1 shows the ECOMP Platform architecture.

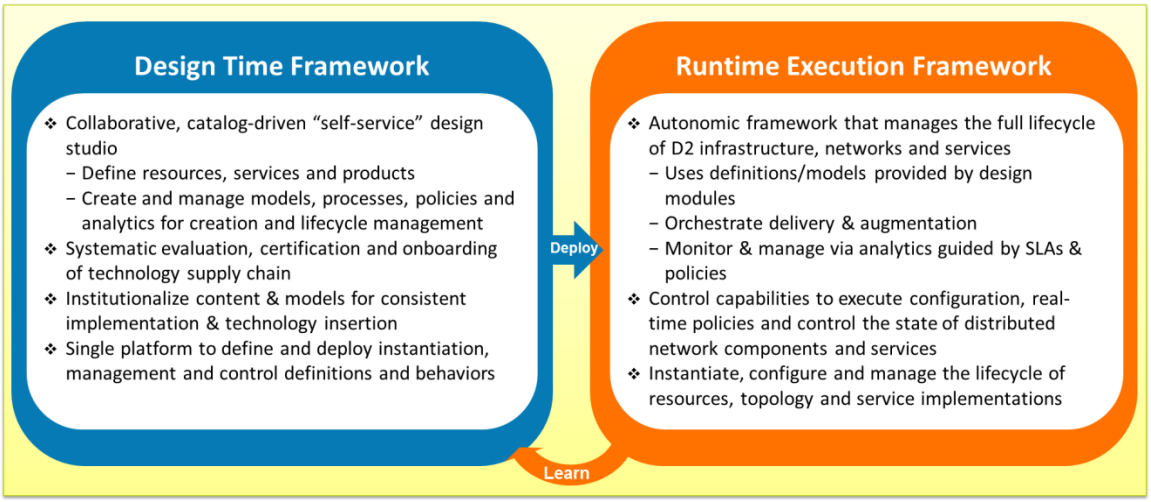

Figure 1: ECOMP Platform

The design time framework component is an integrated development environment with tools, techniques, and repositories for defining/describing AT&T assets. The design time framework facilitates re-use models thus improving efficiency as more and more models are available for reuse. Assets include models of D2 resources, services and products. The models include various process specifications and policies (e.g., rule sets) for controlling behavior and process execution. Process specifications are used by ECOMP to automatically sequence the instantiation, delivery and lifecycle management aspects of D2-based resources, services, products and ECOMP components themselves. The design time framework supports the development of new capabilities, augmentation of existing capabilities and operational improvements throughout the lifecycle of a service. ASDC, Policy, and Data Collection, Analytics and Events (DCAE) SDKs allow operations/security, 3rd parties (e.g., vendors), and other experts to continually define/refine new collection, analytics, and policies (including recipes for corrective/remedial action) using the ECOMP Design Framework Portal. Certain process specifications (aka 'recipes') and policies are geographically distributed to many points of use to optimize performance and maximize autonomous behavior in D2's federated cloud environment. Figure 2 provides a high-level view of

the ECOMP Platform components. These components use micro-services to perform their roles. The Platform provides the common functions (e.g., data collection, control loops, meta-data recipe creation, policy/recipe distribution, etc.) necessary to construct specific behaviors. To create a service or operational capability, it is necessary to develop service/operations-specific collection, analytics, and policies (including recipes for corrective/remedial action) using the ECOMP Design Framework Portal.

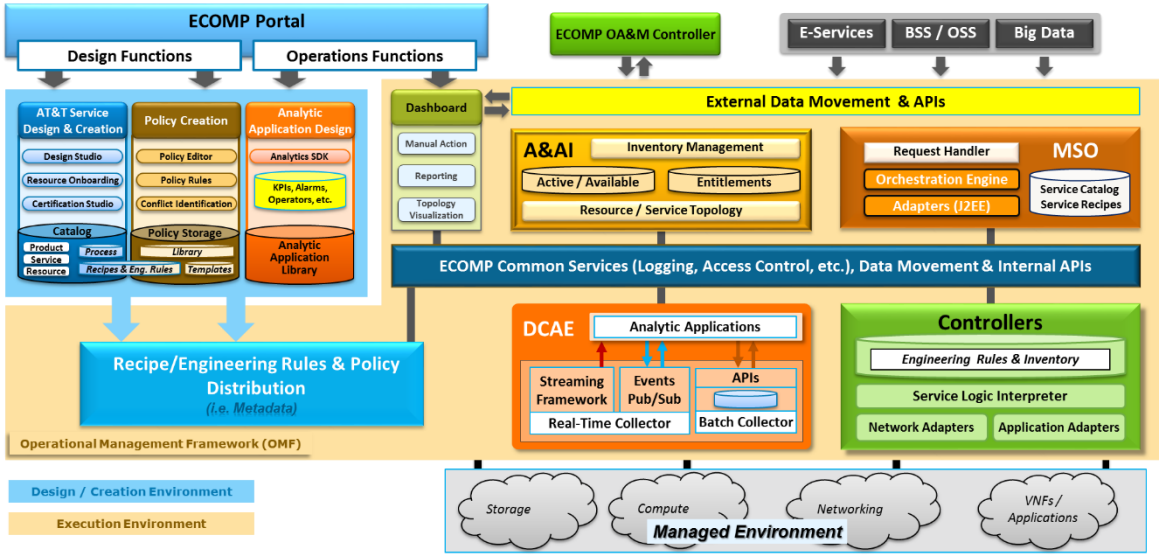

Figure 2: ECOMP Platform components

The two primary components of the design time framework are AT&T Service Design and Creation (ASDC) and the Policy Creation components. ASDC is an integrated development environment with tools, techniques, and repositories to define/simulate/certify D2 assets as well as their associated processes and policies. Each asset is categorized into one of four (4) asset groups: Resource, Services, Products, or Offers. The Policy Creation component deals with Polices; these are conditions, requirements, constraints, attributes, or needs that must be provided, maintained, and/or enforced. At a lower level, Policy involves machine-readable rules enabling actions to be taken based on triggers or requests. Policies often consider specific conditions in effect (both in terms of triggering specific policies when conditions are met, and in selecting specific outcomes of the evaluated policies appropriate to the conditions). Policy allows rapid updates through easily updating rules, thus updating technical behaviors of components in which those policies are used, without requiring rewrites of their software code. Policy permits simpler management / control of complex mechanisms via abstraction.

The design and creation environment supports a multitude of diverse users via common services and utilities. Using the design studio, Product and Service designers onboard/extend/retire resources, services and products. Operations, Engineers, Customer Experience Managers, and Security Experts create workflows, policies and methods to implement Closed Loop Automation and manage elastic scalability.

The runtime execution framework executes the rules and policies distributed by the design and creation environment. This allows us to distribute policy enforcement and templates among various ECOMP modules such as the Master Service Orchestrator (MSO), Controllers, Data Collection, Analytics and Events (DCAE), Active and Available Inventory (A&AI), and a Security Framework. These components advantageously use common services that support logging, access control, and data management.

Orchestration is the function defined via a process specification that is executed by an orchestrator component which automates sequences of activities, tasks, rules and policies needed for on-demand creation, modification or removal of network, application or infrastructure services and resources. The MSO provides orchestration at a very high level, with an end to end view of the infrastructure, network, and application scopes. Controllers are

applications which are intimate with cloud and network services and execute the configuration, real-time policies, and control the state of distributed components and services. Rather than using a single monolithic control layer, AT&T has chosen to use three distinct Controller types that manage resources in the execution environment corresponding to their assigned controlled domain such as cloud computing resources (Infrastructure Controller, typically within the cloud layer), network configuration (Network Controller) and application (Application Controller).

DCAE and other ECOMP components provide FCAPS (Fault Configuration Accounting Performance Security) functionality. DCAE supports closed loop control and higher-level correlation for business and operations activities. It is the ecosystem component supporting analytics and events: it collects performance, usage, and configuration data; provides computation of analytics; aids in trouble-shooting; and publishes events, data and analytics (e.g., to policy, orchestration, and the data lake).

A&AI is the ECOMP component that provides real-time views of Domain 2.0 Resources, Services, Products and their relationships. The views provided by Active and Available Inventory relate data managed by multiple ECOMP Platforms, Business Support Systems (BSS), Operation Support Systems (OSS), and network applications to form a "top to bottom" view ranging from the Products customers buy to the Resources that form the raw material for creating the Products. Active and Available Inventory not only forms a registry of Products, Services, and Resources, it also maintains up-to-date views of the relationships between these inventory items. To deliver the vision of the dynamism of Domain 2.0, Active and Available Inventory will manage these multi-dimensional relationships in real-time.

Active and Available Inventory is updated in real-time by controllers as they make changes in the Domain 2 environment. A&AI is metadata driven, allowing new inventory item types to be added dynamically and quickly via ASDC catalog definitions, eliminating the need for lengthy development cycles.

The platform includes a real-time dashboard, controller and administration tools to monitor and manage all the ECOMP components via an OA&M (Operations, Administration & Management) instance of ECOMP. It allows the design studio to onboard ECOMP components and create recipes, and it allows the policy framework to define ECOMP automation.

ECOMP delivers a single, consistent user experience based on the user's role and allows D2 role changes to be configured within the single ecosystem. This user experience is managed by the ECOMP Portal. The ECOMP Portal provides access to design, analytics and operational control/administration functions via a common role based menu or dashboard. The portal architecture provides web based capabilities including application onboarding and management, centralized access management, dashboards as well as hosted application widgets. The portal provides an SDK to drive multiple development teams to adhere to consistent UI development requirements by taking advantage of built-in capabilities (Services/ API/ UI controls), tools and technologies.

ECOMP provides common operational services for all ECOMP components including activity logging, reporting, common data layer, access control, resiliency, and software lifecycle management. These services provides access management and security enforcement, data backup, restoration and recovery. They support standardized VNF interfaces and guidelines.

The virtual operating environment of D2 introduces new security challenges and opportunities. ECOMP provides increased security by embedding access controls in each ECOMP platform component, augmented by analytics and policy components specifically designed for the detection and mitigation of security violations.

3. ETSI – NFV MANO and ECOMP Alignment

The European Telecommunications Standards Institute (ETSI) developed a Reference Architecture Framework and specifications in support of NFV Management and Orchestration (MANO). The main components of the ETSI-NFV architecture are: Orchestrator, VNF Manager, and VI (Virtualized Infrastructure) Manager. ECOMP expands the scope of ETSI MANO coverage by including Controller and Policy components. Policy plays an important role to control and manage behavior of various VNFs and the management framework. ECOMP also significantly increases the scope of ETSI MANO's resource description to include complete meta-data for lifecycle management of the virtual environment (Infrastructure as well as VNFs). The ECOMP design framework is used to create resource / service /

product definitions (consistent with MANO) as well as engineering rules, recipes for various actions, policies and processes. The Meta-data-driven Generic VNF manager (i.e., ECOMP) allows us to quickly on-board new VNF types, without going through long development and integration cycles and efficiently manage cross-dependencies between various VNFs. Once a VNF is on-boarded, the design time framework facilitates rapid incorporation into future services.

ECOMP can be considered as an *enhanced* Generic VNF manager as described by MANO NFV Management and Orchestration Architectural Options (see Figure 3).

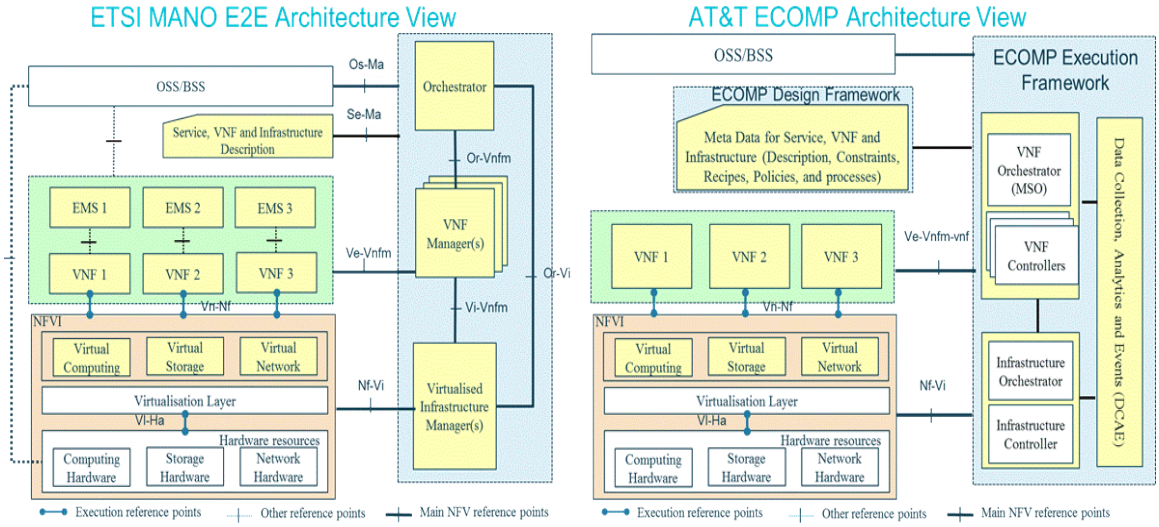

Figure 3: Comparison of ETSI MANO and AT&T ECOMP Architectures

In addition, ECOMP subsumes the traditional FCAPS functionality as supported by EMSs in the MANO reference architecture. This is critical to implementing analytic driven closed loop automation across the infrastructure and VNFs, analyzing cross dependencies and responding to the root cause of problems as quickly as possible. To successfully implement ECOMP, the Ve-Vnfm-vnf interface (as well as Nf-Vi) is critical. AT&T expects the Ve-Vnfm-vnf interface to be a standard interface defined by ETSI. AT&T's approach is to document detailed specifications for such interface(s) to collect rich real-time data in a standard format from a variety of VNFs and quickly integrate with ECOMP without long custom development work.

4. Metadata Driven Design Time and Runtime Execution

Metadata is generally described as "data about data" and is a critical architecture concept dealing with both abstraction and methodology. Metadata expresses structural and operational aspects of the virtualized elements comprising products, services, and resources as expressed in terms of logical objects within a formally defined model space. The attributes of these objects, and the relationships among them, embody semantics that correspond to real-world-aspects of the modeled elements. The modeling process abstracts common features and internal behaviors in order to drive architectural consistency and operational efficiency. The logical representations of underlying elements can be manipulated and extended by designers in consistent ways, and uniformly consumed by the runtime execution framework. Alignment between elements in the model space and those in the real world is continuously maintained through tooling that resolves dependencies, handles exceptions, and infers required actions based on the metadata associated with the modeled elements.

One of the key benefits in AT&T's D2 plans for a virtualized network architecture is a significantly decreased time from service concept to market. The need to operationalize on a service-specific basis would be a major obstacle to this goal. Thus, AT&T plans to manage its D2 network and services via the execution environment driven by a common (service independent) operations support model populated with service-specific metadata. In conjunction with

this, these support systems will provide high levels of service management automation through the use of metadata driven event based control loops.

Such an approach achieves another benefit of driving down support systems costs through the ability to support new services with changes only in the metadata, and without any code modifications. In addition, a common operations support model across all AT&T offered services, coupled with high automation, drives down operations support costs.

AT&T is implementing the metadata driven methodology in D2 by centralizing the creation of rules, and policies in ASDC. All ECOMP applications and controllers will ingest the metadata that governs their behavior from ASDC. Implementation of the metadata-driven methodology requires an enterprise-wide paradigm change of the development process. It demands an upfront agreement on the overall metadata-model across the business (e.g., product development) and the software developers writing code for ECOMP, BSS, OSS, and AIC. This agreement is a behavioral contract to permit both the detailed business analysis and the construction of software to run in parallel. The Software development teams focus on building service-independent software within a common operations framework for runtime automation; while the Business teams can focus on tackling unique characteristics of the business needs by defining metadata models to feed the execution environment. The metadata model content itself is the glue between design time and runtime execution frameworks, and the result is that the service-specific metadata models drive the common service-independent software for runtime execution.

The metadata model is managed through a design environment (ASDC) which guides a series of designers from 3rd party resource onboarding through service creation, verification, and distribution. The modular nature of the ASDC metadata model provides a catalog of patterns that can be reused in future services. This provides a rich forward-engineering approach to quickly extend resource functions to manageable services and sellable products, further realizing benefits in time to market.

ASDC is the ECOMP component that supports AT&T's metadata model design. Many different models of metadata exist to address different business and technology domains. ASDC integrates various tools supporting multiple types of data input (e.g., YANG, HEAT, TOSCA, YAML, BPMN/BPEL, etc.). It automates the formation of an AT&T internal metadata format to drive end-to-end runtime execution. ASDC provides a cohesive and collaborative environment for design, test, certification, version control, and distribution of metadata models across the Resource, Service, Product, and Offer development lifecycle.

In ASDC the resource metadata is created in the description of the cloud infrastructure (e.g., AIC requirements) and the configuration data attributes to support the implementations. Subsequently, the resource metadata descriptions become a pool of building blocks that may be combined in developing services, and the combined service models to form products.

In addition to the description of the object itself, modeling the management *needs* of an object using metadata patterns in ASDC may be applied to almost any business and operations functions in AT&T. For example, metadata can be used to describe the mapping of resource attributes to relational tables, through the definition of rules around the runtime session management of a group of related resources to the signatures of services offered by a particular type of resources. Such rules form the policy definitions that can be used to control the underlying behavior of the software function. Another example is to describe the workflow steps as a process in the metadata that can be used by ECOMP Orchestration to fulfill a customer service request.

D2 policies are one type of metadata, and will eventually be quite numerous and support many purposes. Policies are created and managed centrally so that all policy actions, both simple and complex, are easily visualized and understood together, and validated properly prior to use. Once validated and corrected for any conflicts, the policies are precisely distributed to the many points of use/enforcement; as well, the decisions and actions taken by policy are distributed, but are still part of the policy component of ECOMP. In this manner, policies will already be available when needed by a component, minimizing real-time requests to a central policy engine / PDP (Policy Decision Point), or to Policy Distribution. This improves scalability and reduces latency.

ASDC and Policy Creation exist in close relationship. Policies are created by many user groups (e.g., service designers, security personnel, operations staff, etc.). Various techniques are used to validate newly created policies and to help identify and resolve

potential conflicts with pre-existing policies. Validated policies are then stored in repositories. Subsequently, policies are distributed in two ways: (1) Service-related policies are initially distributed in conjunction with the distribution of recipes (created via ASDC), e.g., for service instantiation, and (2) other policies (e.g., some security and operations policies) are unrelated to particular services, and therefore, are independently distributable. In any case, policies are updatable at any time as required.

5. Closed-Loop Automation

Given the D2 vision described above, we expect network elements and services to be instantiated by customers and providers in a significantly dynamic process with real-time response to actionable events. In order to design, engineer, plan, bill and assure these dynamic services, we have three (3) major requirements:

- A robust design framework that allows specification of the service in all aspects – modeling the resources and relationships that make up the service, specifying the policy rules that guide the service behavior, specifying the applications, analytics and closed-loop events needed for the elastic management of the service
- An orchestration and control framework (MSO and Controllers) that is recipe/policy driven to provide automated instantiation of the service when needed and managing service demands in an elastic manner
- An analytic framework that closely monitors the service behavior during the service lifecycle based on the specified design, analytics and policies to enable response as required from the control framework, to deal with situations ranging from those that require healing to those that require scaling of the resources to elastically adjust to demand variations.

The following sections describe the ECOMP frameworks designed to address these major requirements. The key pattern that these frameworks help automate is

Design -> Create -> Collect -> Analyze -> Detect -> Publish -> Respond.

We refer to this automation pattern as "Closed-loop automation" in that it provides the necessary automations in proactively responding to network and service conditions without human intervention. A high-level schematic of the "Closed-loop automation" and the various phases within the service lifecycle using the automation is depicted in Figure 4.

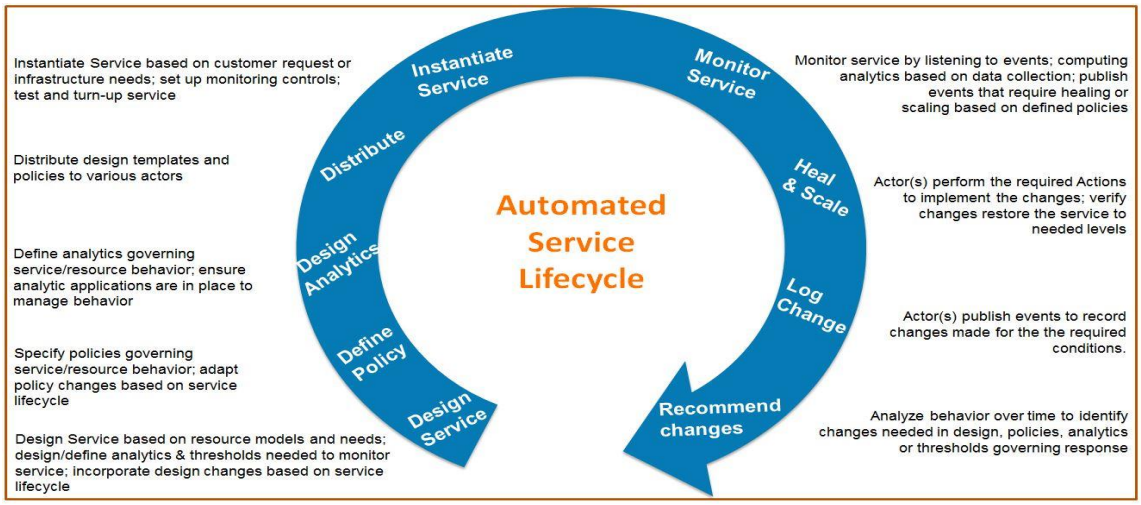

Figure 4: ECOMP Closed Loop Automation

The various phases shown in the service lifecycle above are supported by the Design, Orchestration & Analytic frameworks described below.

Design Framework

The service designers and operations users must, during the design phase, create the necessary recipes, templates and rules for instantiation, control, data collection and analysis functions. Policies and their enforcement points (including those associated with the closed loop automation) must also be defined for various service conditions that require a response, along with the actors and their roles in orchestrating the response. This upfront design ensures that the logic/rules/metadata is codified to describe and manage the closed-loop behavior. The metadata (recipes, templates and policies) is then distributed to the appropriate Orchestration engines, Controllers, and DCAE components.

Orchestration and Control Framework

Closed Loop Automation includes the service instantiation or delivery process. The orchestration and control framework provides the automation of the configuration (both the initial and subsequent changes) necessary for the resources at the appropriate locations in the network to ensure smooth operation of the service. For closed loop automation to occur during the lifecycle management phase of the service, the instantiation phase must ensure that the inventory, monitoring and control functions are activated, including the types of closed loop control related to the participating virtual functions and the overall service health.

Figure 5 illustrates a Runtime Execution of a Service Instantiation high-level use case. As requests come into ECOMP (❶), whether they are customer requests, orders, or an internal operation triggering a network build out, the Orchestration framework will first decompose the request and retrieve the necessary recipe(s) for execution.

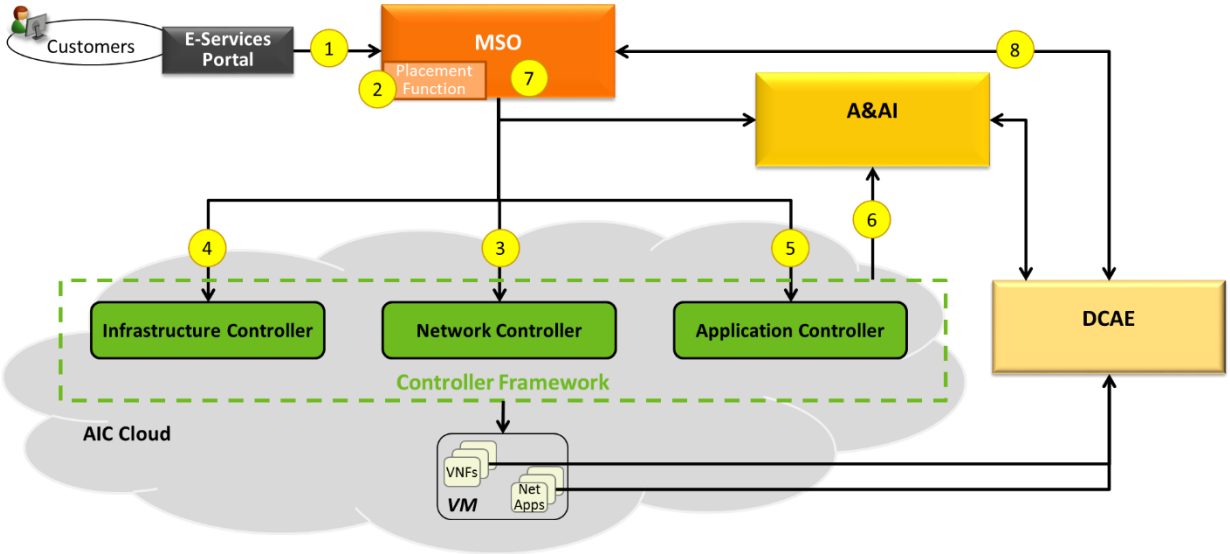

Figure 5: Service Instantiation Use Case

The initial steps of the recipes include a homing and placement task (❷) using constraints specified in the requests. 'Homing and Placement' are micro-services involving orchestration, inventory, and controllers responsible for infrastructure, network, and application. The goal is to allow algorithms to use real-time network data and determine the most efficient use of available infrastructure capacity. Micro-services are policy driven. Examples of policy categories may include geographic server area, LATA/regulatory restrictions, application latency, network resource and bandwidth, infrastructure and VNF capacity, as well as cost and rating parameters.

When a location is recommended and the assignment of resources are done, Orchestration then triggers the various controllers (❸) to create the internal datacenter and WAN networks (L2 VLANs or L3 VPNs), spin up the VMs (❹), load the appropriate VNF software images, connect them to the designated

data plane and control plane networks. Orchestration may also instruct the controllers to further configure the virtual functions for additional L3 features and L4+ capabilities (❺). When the controllers make the changes in the network, autonomous events from the controllers and the networks themselves are emitted for updating the active inventory (❻). Orchestration (MSO) completes the instantiation process by triggering the 'Test and Turn-Up tasks (❼), including the ability for ECOMP to collect service-related events (❽) and policy-driven analytics that trigger the appropriate closed loop automation functions. Similar orchestration/controller recipes and templates as well as the policy definitions are also essential ingredients for any closed loop automation, as Orchestration and Controllers will be the actors that will execute the recommended actions deriving from a closed loop automation policy.

Analytic Framework

The analytic framework ensures that the service is continuously monitored to detect both the anomalous conditions that require healing as well as the service demand variations to enable scaling (up or down) the resources to the right level.

Once the Orchestration and Control framework completes the service instantiation, the DCAE Analytic framework begins to monitor the service by collecting the various data and listening to events from the virtual functions and their agents. The framework processes the data, analyzes the data, stores them as necessary for further analysis (e.g., establishing a baseline, perform trending, look for a signature) and provides the information to the Application Controller. The applications in the framework look for specific conditions or signatures based on the analysis. When a condition is detected, the application publishes a corresponding event. The subsequent steps will depend on the specific policies associated with the condition. In the most simple case, the orchestration framework proceeds to perform the changes necessary (as defined by the policy and design for the condition) to alleviate the condition. In more complex cases, the actor responsible for the event would execute the complex policy rules (defined at design time) for determining the response. In other cases, where the condition does not uniquely identify specific response(s), the responsible actor would conduct a series of additional steps (defined by recipes at design time, e.g., running a test, query history) to further analyze the condition. The results of such diagnosis might further result in publishing of a more specific condition for which there is a defined response. The conditions referred to here could be ones related to the health of the virtualized function (e.g., blade down, software hung, service table overflow, etc.) that require healing. The conditions could be related to the overall service (not a specific virtual function, e.g., network congestion) that requires healing (e.g., re-route traffic). The conditions could also relate to capacity conditions (e.g., based on service demand variation, congestion) resulting in a closed-loop response that appropriately scales up (or down) the service. In the cases where anomalous conditions are detected but specific responses are not identifiable, the condition will be represented in an Operations portal for additional operational analysis.

The analytic framework includes applications that analyze the history of the service lifecycle to discern patterns governing usage, thresholds, events, policy effectiveness etc. and enable the feedback necessary to effect changes in the service design, policies or analytics. This completes the service lifecycle and provides an iterative way to continuously evolve the service to better utilization, better experience and increased automation.

6. ASDC

ASDC is the component within the design time environment that provides multiple organizations the ability to create and manage AT&T D2 assets in terms of "models". ASDC asset models are generally categorized into four object types: Resource, Service, Product and Offer.

A Resource model represents a fundamental capability of D2 which is developed by AT&T or a 3rd party supplier. Resources, either hardware or software, can be categorized as:

- Infrastructure Resource – (the Cloud resources, e.g., Compute, Storage).
- Network Resource (network connectivity functions & elements).
- Application Resource (the features and capabilities of a software application).

A Service model represents a well-formed object with one or more resources (compute + connectivity + app functions/features) composed and operationalized in AT&T environment. In some cases a Service supports

multiple customers, while in other cases a Service will be dedicated to a single customer.

A Product model includes one or more services packaged with base commercialization attributes for customer ordering and billing of the underlying service(s). An Offer model specifies the bundling of products with specific Marketing configurations for selling to the customers.

Figure 6 provides a high level overview of the aspects of the metadata-driven model methodology using ASDC Models.

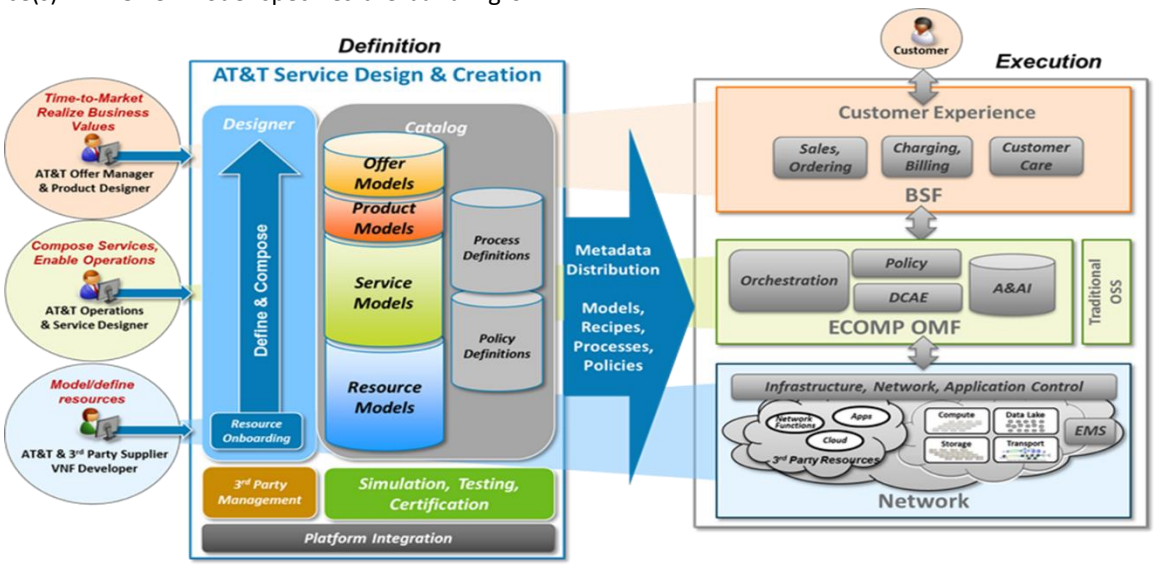

Figure 6: ASDC Meta-Driven Methodology

The specific models stored in the ASDC Master Reference Catalog are reusable by the entire enterprise. The content from the Master Catalog is distributed to the runtime "execution engines" so they can efficiently interoperate and consistently execute.

This model driven approach allows AT&T to vastly reduce the time needed to bring new products, services, and resources to market, and enables AT&T to open its Network to partners and industry developers for rapid on-boarding of 3rd party solutions.

6.1 ASDC Model Framework

A model in ASDC is the profile of an asset item expressed with its *Descriptor* and *Management Recipes*. ASDC provides basic templates for modeling Resources, Services, and Products. They are all derived from a generic extendable template as depicted in Figure 7.

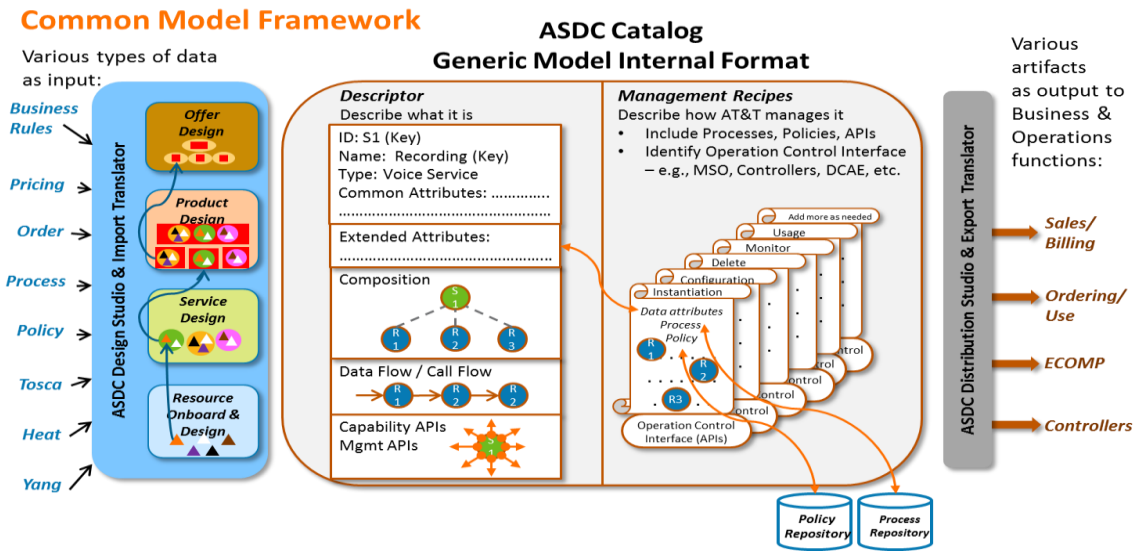

Figure 7: Common Model Framework

The *Descriptor* defines the capability of the resource, service, or product item provided by the developer/designer. It contains the following information:

- Common Parameters which provide required information across all asset items, such as ID and name.
- Additional parameters (0 to N) that may be added for any information related to a specific asset model.
- Composition graphs (0 to N) that may be included to express the composition & relationship with the underlying components.
- APIs (0 to N) that are exposed by the asset item itself to provide functional or management capabilities.
- Software executables (0 to N).

A *Management Recipe* includes the *Parameters*, *Processes*, and *Policies* related to AT&T's methods of instantiation and management of an actual asset instance. These recipes are fed to AT&T Management Systems via APIs for metadata-driven execution, with each recipe targeted for a specific operations or control function. The number of recipes included in a particular asset model is variable (from 0 to N) and determined by the capability of the corresponding execution system for accepting the metadata to drive its execution. For example:

- A "*Configuration Recipe*" may be created to specify the execution process for instantiation, change, and deletion of an asset instance.
  - At the Resource layer, the configuration recipe may be expressed by standards-based syntax and Open Source model (e.g., TOSCA, HEAT for infrastructure, and YANG for the Network) to drive the controller execution.
  - At the Service layer, a Configuration Recipe may be expressed by BPEL instructions to drive the ECOMP MSO execution.
- An "*Ordering Recipe*" at the product layer that may be composed of BPEL instructions to provide the ordering flow executed by an Ordering BSS.
- "*Monitoring Recipe*" for fault, "*Performance Recipe*" for performance analysis, or "*Usage Recipe*" for Usage Measurement at the service layer that may be used by ECOMP DCAE or other service assurance OSS's.

6.2 ASDC Functional Architecture

ASDC employs a set of "studios" to design, certify, and distribute standardized models including the relationships within and across them. From a bottom-up view, ASDC provides tools to onboard resources such as building blocks and make them available for enterprise-wide composition. From a top-down view, ASDC supports product managers who compose new products from existing services/resources in the ASDC catalog or acquire new resource capabilities from internal or external developers. The components of ASDC are shown in Figure 8.

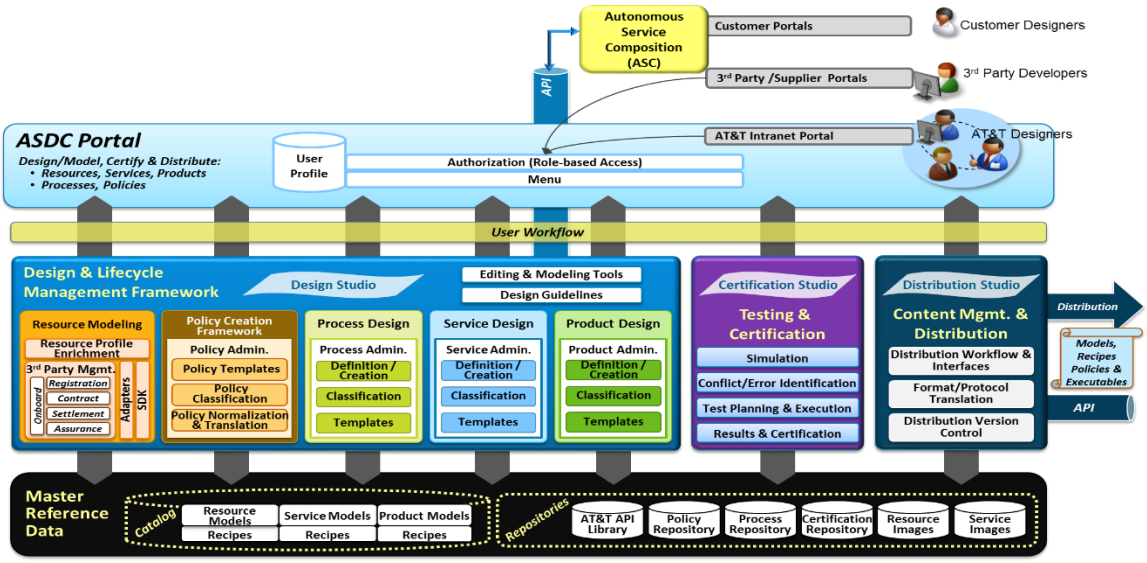

Figure 8: Design Studio

The ASDC "Design Studio" consists of a portal and back end tooling for onboarding, iterative modeling, and validation of AT&T assets. The Design Studio includes a set of basic model templates that each project can further configure and extend with additional parameters, parameter value ranges and validation rules. The configured project-specific templates are used by the Design Studio GUI (as drop-down menu or rule-based validation) to validate the designs. This ensures the models contain the necessary information and valid values based on the type of model and the project requirements.

ASDC provides access to modeling tools that can be used to create executable process definitions that will be used by the various D2 components. Processes can be created with standard process modeling tools such as BPMN (Business Process Management Notation) tools. The processes created are stored in the Process Repository and asset models in the catalog may refer to them.

The models in the Master Reference Catalog can be translated to any industry-standard or required proprietary format by the "Distribution Studio". The modeling process does not result in any instantiation in the run-time environment until the MSO receives a request to do so.

ASDC will integrate with AT&T's Third Party Supplier Management functions as needed for onboarding, modeling, and cataloging software assets along with their entitlement and license models as part of the Design Studio.

Data Repositories

ASDC Data Repositories maintain the design artifacts and expose content to the designers, testers, and distributors. The repositories include:

Master Reference Catalog is the data store of designed models, including resources, services, and products, the relationships of the asset items, and their references to the process & policy repositories.

Process Repository is the data store of designed processes.

Policy Repository is the data store of designed policies.

Resource Images is the data store that contains the resource executables.

Service Images is the data store that contains the executables for the service.

Certification Repository is the data store of testing artifacts.

Certification Studio

ASDC provides a Certification Studio with expanded use of automated simulation and test tools along with access to a shared, virtualized testing sandbox. The model-driven approach to testing enables a reduction in overall deployment cost, complexity, and cycle time. The studio:

- Allows reuse and reallocation of hardware resources as needed for testing rather than dedicated lab environments
- Supports test environment instantiation of various sizes when needed using a mix of production and test components, using the standardized and automated D2 operations framework
- Provides expanded use of automated test tools beginning in design through deployment for simulation/modeling behaviors, conflict and error identification, test planning and execution, and results/certification

Distribution Studio

The D2 models are stored in an internal AT&T technology independent format which provides AT&T greater flexibility in selecting the D2 execution engines that consume the data. In a model-driven, software-based architecture, controlling the distribution of model data and software executables is critical. This Studio provides a flexible, auditable mechanism to format, translate when needed, and distribute the models to the various D2 components. Validated models are distributed from the design time environment to a runtime repository. The runtime Distribution component supports two modes of access: 1) models can be sent to the component using the model, in advance of when they are needed or 2) models can be accessed in real-time by the runtime components. This distribution is intelligent, such that each function automatically receives or has access to only the specific model components which match its needs and scope.

7. Policy

The Policy platform plays an important role in realizing the D2.0 vision of closed loop automation and lifecycle management. The Policy platform's main objective is to control/affect/modify behavior of the complete D2.0 Environment (NFVI, VNF, ECOMP, etc.) using field configurable policies/rules without always requiring a development cycle. Conceptually, "Policy" is an approach to intelligently constrain and/or influence the behaviors of functions and systems. Policy permits simpler management/ control of complex mechanisms via abstraction. Incorporating high-level goals, a set of technologies and architecture, and supportive methods/patterns, Policy is based on easily-updateable conditional rules, which are implemented in various ways such that policies and resulting behaviors can be quickly changed as needed. Policy is used to control, to influence, and to help ensure compliance with goals.

"A policy" in the sense of a particular D2.0 policy may be defined at a high level to create a condition, requirement, constraint, or need that must be provided, maintained, and enforced. A policy may also be defined at a lower or "functional" level, such as a machine-readable rule or software condition/assertion which enables actions to be taken based on a trigger or request, specific to particular selected conditions in effect at that time. This can include XACML policies, Drool policies, etc. lower level policies may also be embodied in models such as YANG, TOSCA, etc.

7.1 Policy Creation

The ECOMP policy platform has a broad scope supporting infrastructure, product / services, operation automation, security-related policy rules. These policy rules are defined by multiple stakeholders, (Network / Service Designers, Operations, Security, customers, etc.). In addition, input from various sources (ASDC, Policy Editor, Customer Input, etc.) should be collected and rationalized. Therefore, a centralized policy creation environment will be used to validate policies rules, identify and resolve overlaps & conflicts, and derive policies where needed. This creation framework should be universally accessible, developed and managed as a common asset, and provides editing tools to allow users to easily create or change policy rules. Offline analysis of performance/fault/closed-loop action data are used to identify opportunities to discover new signatures and refine existing signatures and closed loop operations. Policy translation/ derivation functionality is also included to derive lower level policies from higher level policies. Conflict detection and mitigation are used to detect and resolve policies that may potentially cause conflicts, prior to distribution. Once validated and free of conflicts, policies are placed in an appropriate repository.

7.2 Policy Distribution

After completing initial policy creation or modification to existing policies, the Policy Distribution Framework sends policies (e.g., from the repository) to their points of use, *in advance* of when they are needed. This distribution is intelligent and precise, such that each distributed policy-enabled function automatically receives only the specific policies which match its needs and scope.

Notifications or events can be used to communicate links/URLs for policies to components needing policies, so that components can utilize those links to fetch particular policies or groups of policies as needed. Components in some cases may also publish events indicating they need new policies, eliciting a response with updated links/URLs. Also, in some cases policies can be given to components indicating they should *subscribe* to one or more policies, so that they receive updates to those policies automatically as they become available.

7.3 Policy Decision and Enforcement

Runtime policy decision and enforcement functionality is a distributed system that can apply to the various ECOMP modules in most cases (there could be some exceptions). For example, Policy rules for data collection and their frequency are enforced by DCAE data collection functionality. Analytic policy rules, anomalous / abnormal condition identification, and publication of events signaling detection of such conditions are enforced by DCAE Analytic applications. Policy rules for associated remedial or other action (e.g., further diagnosis) are enforced by the right actor/participant in a control loop (MSO, Controller, DCAE, etc.).

Policy Decision/Enforcement functionality generally receives policies in advance, via Policy Distribution. In some cases a particular runtime Policy engine may be queried in real-time for policies/guidance, as indicated in the previous section. Additional unifying mechanisms, methods, and attributes help manage complexity and ensure that Policy is not added inefficiently as separate "islands." Attribute values may be defined at creation time. Examples include Policy Scope attributes, described in the following section ("Policy Unification and Organization"). Note also that Policy objects and attributes will need to be included in a proper governance process to ensure that correct intended outcomes are achieved for the business.

Policy related APIs can provide the ability to: 1. *Get* (read) policies from a component, i.e., on demand, 2. *Set* (write) one or more policies into a component, i.e., immediately pushed/updated, and 3. *Distribute* a set of policies to multiple components that match the scope of those policies, for immediate use (forced) or later use (upon need, e.g., time-determined) by those entities.

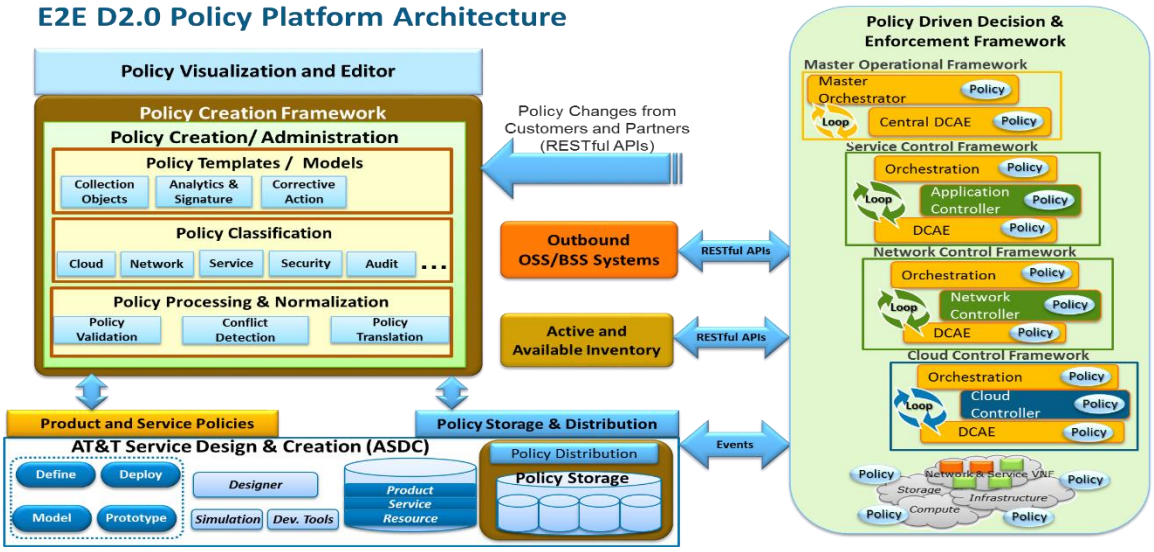

Figure 9: D2 Policy Architecture Framework

Figure 9 shows Policy Creation on the left, Policy Repository & Distribution at the bottom, and Policy use on the right (e.g., in Control Loops, or in VNFs). As shown in Figure 9, Policy Creation is in close association with ASDC. When fully integrated, policies will be created either in conjunction with Products & Services (for policy scopes that are related to these), or separately for policies of scopes orthogonal to these (i.e., unrelated to particular products & services). Orthogonal policies may include various policies for operations, security, infrastructure optimization, etc.

Note that the architecture shown is a logical architecture, and may be implemented in various ways. Some functions in whole or in part may be implemented either as separate virtualized elements or within other (non-policy) functions.

7.4 Policy Unification and Organization

In an expandable, multi-purpose Policy Framework, many types of Policy may be used. Policy may be organized using many convenient dimensions in order to facilitate the workings of the Framework within D2.0. A flexible organizing principle termed Policy Scope will enable a set of attributes to specify (to the degree/precision desired, and using any set of desired "dimensions") the precise "scope" of both policies and policy-enabled functions/components. Useful organizing *dimensions* of Policy Scope may include: (a) Policy type or category, e.g., taxonomical, (b) Policy ownership / administrative domain, (c) geographic area or location, (d) technology type and/or specifics, (e) Policy language, version, etc., (f) security level or other security-related values/specifiers/limiters, (g) particular defined grouping, and (h) any other dimensions/attributes as may be deemed helpful, e.g., by Operations. Note that attributes can be defined for each dimension.

By then setting values for these attributes, Policy Scope can be used to specify the precise Policy "scope" of: (A) Policy events or requests/triggers to allow each event/request to self-indicate its scope, e.g., which can then be examined by a suitable function for specifics of routing/delivery, (B) Policy decision/enforcement functions or other Policy functions to allow each Policy function to self-indicate its scope of decision making, enforcement, or other capabilities, (C) Virtual Functions of any type for auto-attachment to the appropriate Policy Framework and distribution mechanism instances, and most importantly to (D) individual policies to aid in management and distribution of the policies.

7.5 Policy Technologies

D2 Policy will utilize rather than replace various technologies; examples of possible policy areas are shown in the following table. These will be used, e.g., via translation capabilities, to achieve the best possible solution that takes advantage of helpful technologies while still providing in effect a single D2.0 Policy "brain."

Policy Technologies and Their Scopes (to be unified & coordinated)

Technology	Description	(Initial) Scope
Policy Applications	Plug-ins (in some fashion) to the Policy Platform, providing needed functions.	Additional functionality, e.g. for Conflict Detection
XACML++	XACML 3.0, extended for purposes beyond XACML's traditional access control focus.	1. Overarching/core policies 2. Policies not handled by DS technologies
OpenStack Congress	OpenStack policy-as-a-service	Detect policy violation for OpenStack resources
OpenStack Heat	OpenStack cloud orchestration	Resource orchestration policies, delegated
OpenStack GBP	Group-Based Policy (GBP) for OpenStack-managed resources	Neutron (networking), followed by Nova (compute), Swift (storage), etc
OpenDaylight GBP	Group-Based Policy (GBP) for network resources under SDN control	Service Function Chaining (SFC)
ASTRA	Policy-enabled firewall control, etc.	Security policies, firewall etc., delegated.
IAM / IDAM	Identity and Access Management	Security policies, Identity/Access, delegated.
YANG / TOSCA	Modeling approaches, SDN and higher level	Significant portion of SDN control
Drools	Business rules management system	Attribute/model based rule evaluation
RUBY	AT&T "rules you build yourself" project/capability	Domain 1.0 policies, leveraged

7.6 Policy Use

At runtime, policies that were previously distributed to policy-enabled components will be used by those components to control or influence their functionality and behavior, including any actions that are taken. In many cases, those policies will be utilized to make decisions, where these decisions will often be conditional upon the current situation.

A major example of this approach is the feedback/control loop pattern driven by DCAE. Many

specific control loops can be defined. In a particular control loop, each participant (e.g., orchestrator, controller, DCAE, virtual function) will have received policies determining how it should act as part of that loop. All of the policies for that loop will have been previously created together, ensuring proper coordinated closed-loop action. DCAE can receive specific policies for data collection (e.g., what data to collect, how to collect, and how often), data analysis (e.g., what type and depth of analysis to perform), and signature and event publishing (e.g., what analysis results to look for as well as the specifics of the event to be published upon detecting those results). Remaining components of the loop (e.g., orchestrators, controllers, etc.) can receive specific policies determining actions to be taken upon receiving the triggered event from DCAE. Each loop participant could also receive policies determining the specific events to which it subscribes.

8. Orchestration

In general, Orchestration can be viewed as the definition and execution of workflows or processes to manage the completion of a task. The ability to graphically design and modify a workflow process is the key differentiator between an orchestrated process and a standard compiled set of procedural code.

Orchestration provides adaptability and improved time-to-market due to the ease of definition and change without the need for a development engagement. As such, it is a primary driver of flexibility in the architecture. Interoperating with Policy, the combination provides a basis for the definition of a flexible process that can be guided by business and technical policies and driven by process designers.

Orchestration exists throughout the D2 architecture and should not be limited to the constraints implied by the term “workflow” as it typically implies some degree of human intervention. Orchestration in D2 will not involve human intervention/decision/ guidance in the vast majority of cases. The human involvement in orchestration is typically performed up front in the design process although there may be processes that will require intervention or alternate action such as exception or fallout processing.

To support the large number of Orchestration requests, the orchestration engine will be exposed as a reusable service. With this approach, any component of the architecture can execute process recipes. Orchestration Services will be capable of consuming a process recipe and executing against it to completion. The Service model maintains consistency and reusability across all orchestration activities and ensures consistent methods, structure and version of the workflow execution environment.

Orchestration Services will expose a common set of APIs to drive consistency across the interaction of ECOMP components. To maintain consistency across the platform, orchestration processes will interact with other platform components or external systems via standard and well-defined APIs.

The Master Service Orchestrator’s (MSO’s) primary function is the automation of end-to-end service instance provisioning activities. The MSO is responsible for the instantiation/release, and migration/relocation of VNFs in support of overall ECOMP end-to-end service instantiation, operations and management. The MSO executes well-defined processes to complete its objectives and is typically triggered by the receipt of ‘service requests’ generated by other ECOMP components or by Order Lifecycle Management in the BSS layer. The orchestration “recipe” is obtained from the Service Design and Creation (ASDC) component of the ECOMP Platform where all Service Designs are created and exposed/distributed for consumption.

Controllers (Infrastructure, Network and Application) participate in service instantiation and are the primary players in ongoing service management, e.g., control loop actions, service migration/scaling, service configuration, and service management activities. Each Controller instance supports some form of orchestration to manage operations within its scope.

Figure 10 illustrates the use of Orchestration in the two main areas: Service Orchestration embodied in the Master Service Orchestrator and Service Control embodied in the Infrastructure, Application and Network Controllers. It illustrates the two major domains of D2 that employ orchestration. Although the objectives and scope of the domains vary, they both follow a consistent model for the definition and execution of orchestration activities.

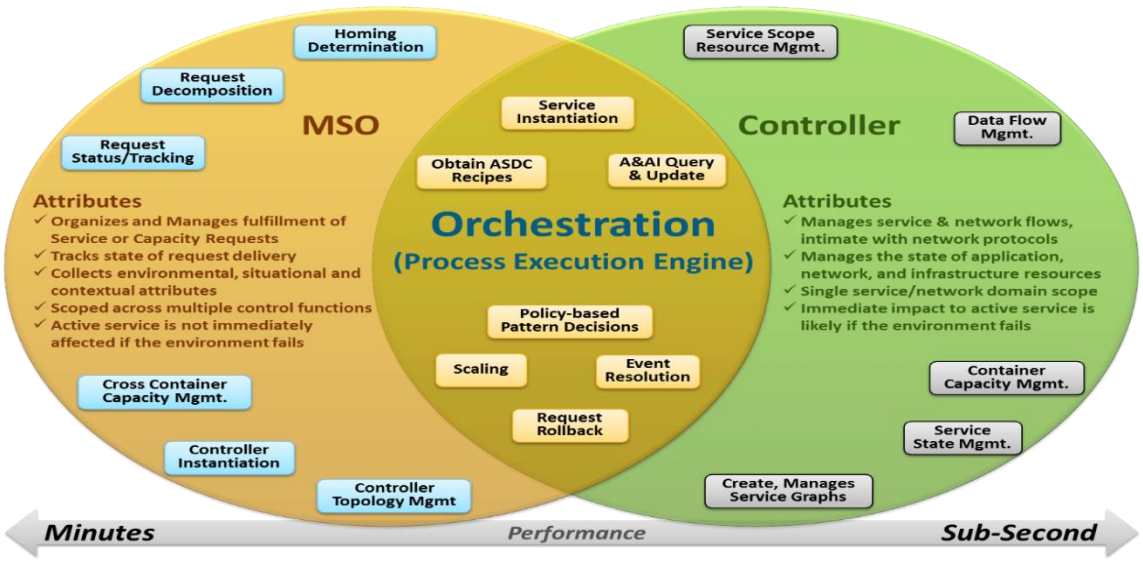

Figure 10: Comparison of MSO and Controllers

Depending on the scope of a network issue, the MSO may delegate, or a Controller may assume, some of the activities identified above. Although the primary orchestrator is called "Master Service Orchestrator" (MSO), its job is to manage orchestration at the top level and to facilitate the orchestration that takes place within the underlying controllers and marshal data between the Controllers such that they have the "process steps" and all the "ingredients" to complete the execution of their respective recipes. For new services, this may involve determination of service placement and identification of existing controllers that meet the Service Request parameters and have the required capacity. If existing controllers (Infrastructure, Network or Application) do not exist or do not have capacity, the MSO will obtain a recipe for instantiation of a new Controller under which the requested Service can be placed.

ASDC is the module of ECOMP where orchestration process flows are defined. These process flows will start with a template that may include common functions such as homing determination, selection of Infrastructure, Network and Application Controllers, consultation of policies and interrogation of A&AI to obtain necessary information to guide the process flows. The MSO does not provide any process-based functionality without a recipe for the requested activity regardless of whether that request is a Customer Order or a Service adjustment/ configuration update to an existing service.

MSO will interrogate A&AI to obtain information regarding existing Network and Application Controllers to support a Service Request. A&AI will provide the addresses of candidate Controllers that are able to support the Service Request. The MSO may then interrogate the Controller to validate its continued available capacity. The MSO and the Controllers report reference information back to A&AI upon completion of a Service request to be used in subsequent operations.

8.1 Application, Network and Infrastructure Controller Orchestration

As previously stated, orchestration is performed throughout the D2 Architecture by various components, primarily the MSO and the Application, Network and Infrastructure controllers. Each will perform orchestration for:

- Service Delivery or Changes to existing Service
- Service Scaling, Optimization, or Migration
- Controller Instantiation
- Capacity Management

Regardless of the focus of the orchestration, all recipes will include the need to update A&AI with configuration information, identifiers and IP Addresses.

Infrastructure Controller Orchestration

Like the MSO, Controllers will obtain their orchestration process and payload (templates/ models) from Service Design & Creation (ASDC). For Service instantiation, the MSO maintains overall end-to-end responsibility for ensuring that a request is completed. As part of that responsibility, the MSO

will select the appropriate controllers (Infrastructure, Network, and Application) to carry out the request. Because a Service Request is often comprised of one or more Resources, the MSO will request the appropriate Controllers to obtain the recipe for the instantiation of a Resource within the scope of the requested Controller. After service placement is determined, the MSO may request the creation of a Virtual Machine (VM) at one or more locations depending on the breadth of the service being instantiated and whether an existing instance of the requested service can be used. If new VM resources are required, the MSO will place the request to the Infrastructure Controller for the specific AIC location. Upon receipt of the request, the Infrastructure Controller may obtain its Resource Recipe from ASDC. The Infrastructure Controller will then begin orchestrating the request. For Infrastructure Controllers, this typically involves execution of OpenStack requests for the creation of virtual machines and for the loading of the Virtual Function (VF) software into the new VM container. The Resource recipe will define VM sizing, including compute, storage and memory. If the Resource Level Recipe requires multiple VMs, the MSO will repeat the process, requesting each Infrastructure Controller to spin up one or more VMs and load the appropriate VFs, again driven by the Resource Recipe of the Infrastructure Controller. When the Infrastructure Controller completes the request, it will pass the virtual resource identifier and access (IP) information back to the MSO to provide to the Network and Application controllers. Along the entire process, the MSO may write identifier information to A&AI for inventory tracking.

Network Controller Orchestration

Network Controllers are constructed and operate in much the same manner as Application and Infrastructure Controllers. New Service requests will be associated with an overall recipe for instantiation of that Service. The MSO will obtain compatible Network Controller information from A&AI and will in turn request LAN or WAN connectivity and configuration to be performed. This may be done by requesting the Network Controller to obtain its resource recipe from ASDC. It is the responsibility of the MSO to request (virtual) network connectivity between the components and to ensure that the selected Network Controller successfully completes the Network configuration workflow. A Service may have LAN, WAN and Access requirements, each of which will be included in the recipe and configured to meet the instance specific customer or service requirements at each level. Physical Access might need to be provisioned in the legacy provisioning systems prior to requesting the MSO to instantiate the service.

Application Control Orchestration

Application Controllers will also be requested by the MSO to obtain the Application Specific component of the Service Recipe from ASDC and execute the orchestration workflow. The MSO continues to be responsible for ensuring that the Application Controller successfully completes its Resource configuration as defined by the recipe. As with Infrastructure and Network Controllers, all workflows, whether focused on Instantiation, configuration or scaling, will be obtained or originate from ASDC. In addition, workflows also will report their actions to A&AI as well as to MSO.

Note that not all changes in network or service behavior are the result of orchestration. For example, application Virtual Functions can change network behavior by changing rules or policies associated with Controller activities. These policy changes can dynamically enable service behavior changes.

9. DCAE

In the D2 vision, virtualized functions across various layers of functionality are expected to be instantiated in a significantly dynamic manner that requires the ability to provide real-time responses to actionable events from virtualized resources, ECOMP applications, as well as requests from customers, AT&T partners and other providers. In order to engineer, plan, bill and assure these dynamic services, DCAE within the ECOMP framework gathers key performance, usage, telemetry and events from the dynamic, multi-vendor virtualized infrastructure in order to compute various analytics and respond with appropriate actions based on any observed anomalies or significant events. These significant events include application events that lead to resource scaling, configuration changes, and other activities as well as faults and performance degradations requiring healing. The collected data and computed analytics are stored for persistence as well as use by other applications for business and operations (e.g., billing, ticketing). More importantly, DCAE has to perform a lot of these functions in real-time. One of the key

design patterns we expect this component to help realize is

"Collect Data → Analyze & Correlate → Detect Anomalies → Publish need for Action".

We expect this pattern in various forms to be realized at multiple layers (e.g., infrastructure, network, and service). We envision the data to be collected once and made available to multiple applications (AT&T, vendors, partners, others) for consumption. We expect applications supporting various operational and business functions to be key consumers of the data & events made available by DCAE. We envision DCAE to be an open analytic framework in allowing AT&T to be able to extend and enhance its capabilities, behavior and scale to support the evolution of the various network functions that are virtualized over time.

9.1 Platform Approach to DCAE

It is essential that we take the ECOMP Platform approach described earlier in order to fulfill the DCAE goals. The notion of Platform here is a reference to the core DCAE capabilities that help define how data is collected, moved, stored and analyzed within DCAE. These capabilities can then be used as a foundation to realize many applications serving the needs of a diverse community. Figure 11 is a functional architecture rendition of the DCAE Platform to enable analytic applications within the ECOMP/DCAE environment.

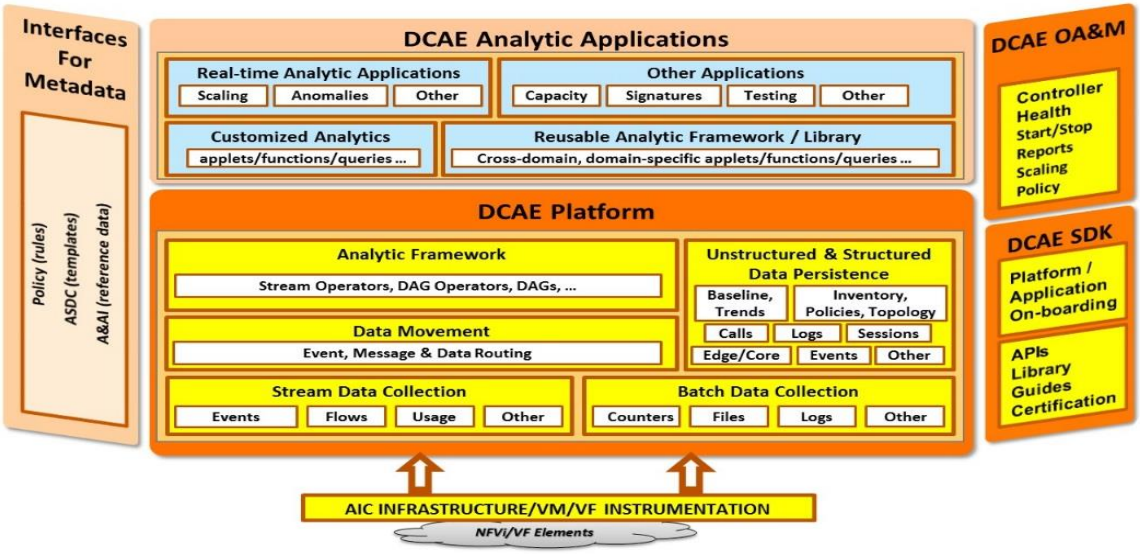

Figure 11: DCAE Platform Approach to Analytic Applications

As Figure 11 suggests, the DCAE Platform requires a development environment with well documented capabilities and toolkit that allows the development and on-boarding of platform components and applications. The DCAE platform and applications depend on the rich instrumentation made available by the underlying AIC infrastructure and the various virtual and physical elements present in that infrastructure to enable the collection, processing, movement and analysis necessary for elastic management of the infrastructure resources. In addition, it relies on robust interfaces with key ECOMP components for the reference information about the managed elements (A&AI), the rules (Policy) and templates (ASDC) that govern the behavior of the managed elements and the ECOMP response.

9.2 DCAE Platform Components

The DCAE Platform consists of multiple components: Common Collection Framework, Data Movement, Edge & Central Lake, Analytic Framework, and Analytic Applications. These are described below:

Common Collection Framework

The collection layer provides the various collectors necessary to collect the instrumentation made available in the AIC infrastructure. The scope of the data collection includes all of the physical and virtual elements (Compute, Storage and Network) in the AIC infrastructure. The collection includes the types of

events data necessary to monitor the health of the managed environment, the types of data to compute the key performance and capacity indicators necessary for elastic management of the resources, the types of granular data (e.g., flow, session & call records) needed for detecting network & service conditions, etc. The collection will support both real-time streaming as well as batch methods of data collection.

Data Movement

This component facilitates the movement of messages and data between various publishers and interested subscribers. While a key component within DCAE, this is also the component that enables data movement between various ECOMP components.

Edge & Central Lake

DCAE needs to support a variety of applications and use cases ranging from real-time applications that have stringent latency requirements to other analytic applications that have a need to process a range of unstructured and structured data. The DCAE storage lake needs to support all of these needs and must do so in a way that allows for incorporating new storage technologies as they become available. This will be done by encapsulating data access via APIs and minimizing application knowledge of the specific technology implementations.

Given the scope of requirements around the volume, velocity and variety of data that DCAE needs to support, the storage will technologies that Big Data has to offer, such as support for NOSQL technologies, including in-memory repositories, and support for raw, structured, unstructured and semi-structured data. While there may be detailed data retained at the DCAE edge layer for detailed analysis and trouble-shooting, applications should optimize the use of precious bandwidth & storage resources by ensuring they propagate only the required data (reduced, transformed, aggregated, etc.) to the Core Data Lake for other analyses.

Analytic Framework

The Analytic Framework is an environment that allows for development of real-time applications (e.g., analytics, anomaly detection, capacity monitoring, congestion monitoring, alarm correlation etc.) as well as other non-real-time applications (e.g., analytics, forwarding synthesized or aggregated or transformed data to Big Data stores and applications); the intent is to structure the environment that allows for agile introduction of applications from various providers (Labs, IT, vendors, etc.). The framework should support the ability to process both a real-time stream of data as well as data collected via traditional batch methods. The framework should support methods that allow developers to compose applications that process data from multiple streams and sources. Analytic applications are developed by various organizations, however, they all run in the DCAE framework and are managed by the DCAE controller. These applications are micro-services developed by a broad community and adhere to ECOMP Framework standards.

Analytic Applications

The following list provides examples of types of applications that can be built on top of DCAE and that depend on the timely collection of detailed data and events by DCAE.

Analytics These will be the most common applications that are processing the collected data and deriving interesting metrics or analytics for use by other applications or Operations. These analytics range from very simple ones (from a single source of data) that compute usage, utilization, latency, etc. to very complex ones that detect specific conditions based on data collected from various sources. The analytics could be capacity indicators used to adjust resources or could be performance indicators pointing to anomalous conditions requiring response.

Fault / Event Correlation This is a key application that processes events and thresholds published by managed resources or other applications that detect specific conditions. Based on defined rules, policies, known signatures and other knowledge about the network or service behavior, this application would determine root cause for various conditions and notify interested applications and Operations.

Performance Surveillance & Visualization This class of application provides a window to Operations notifying them of network and service conditions. The notifications could include outages and impacted services or customers based on various dimensions of interest to Operations. They provide visual aids ranging from geographic dashboards to virtual information model browsers

to detailed drilldown to specific service or customer impacts.

Capacity Planning This class of application provides planners and engineers the ability to adjust forecasts based on observed demands as well as plan specific capacity augments at various levels, e.g., NFVI level (technical plant, racks, clusters, etc.), Network level (bandwidth, circuits, etc.), Service or Customer levels.

Testing & Trouble-shooting This class of application provides operations the tools to test & trouble-shoot specific conditions. They could range from simple health checks for testing purposes, to complex service emulations orchestrated for troubleshooting purposes. In both cases, DCAE provides the ability to collect the results of health checks and tests that are conducted. These checks and tests could be done on an ongoing basis, scheduled or conducted on demand.

Security Some components of AIC may expose new targets for security threats. Orchestration and control, decoupled hardware and software, and commodity hardware may be more susceptible to attack than proprietary hardware. However, SDN and virtual networks also offer an opportunity for collecting a rich set of data for security analytics applications to detect anomalies that signal a security threat, such as DDoS attack, and automatically trigger mitigating action.

Other We note that the applications that are listed above are by no means exhaustive and the open architecture of DCAE will lend itself to integration of application capabilities over time from various sources and providers.

10. Active & Available Inventory (A&AI)

Active and Available Inventory (A&AI) is the ECOMP component that provides real-time views of D2 Resources, Services, Products, and Customer Subscriptions for D2 services. Figure 12 provides a functional view of A&AI. The views provided by Active and Available Inventory relate data managed by multiple ECOMP, BSS, OSS, and network applications to form a "top to bottom" view ranging from the Products customers buy to the Services and Resources used to compose the Products. Active and Available Inventory not only forms a registry of Products, Services, and Resources, it also maintains up-to-date views of the relationships between these inventory items across their lifecycles. To deliver the vision of the dynamism of D2, Active and Available Inventory will manage these multi-dimensional relationships in real-time.

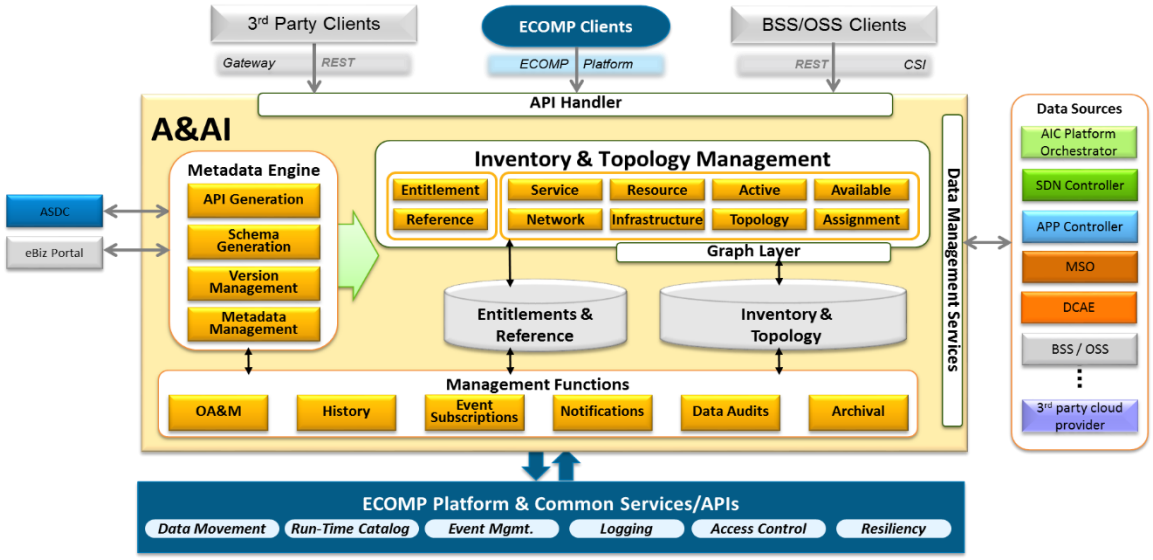

Figure 12: A&AI Functional View

Active and Available Inventory maintains real-time Inventory and Topology data by being continually updated as changes are made within the AT&T Integrated Cloud. It uses graph data technology to store relationships between inventory items. Graph traversals can then be used to identify chains of dependencies between items. A&AI data views are used by homing logic during real-time service

delivery, root cause analysis of problems, impact analysis, capacity management, software license management and many other D2 functions.

The Inventory and Topology data includes resources, service, products, and customer subscriptions, along with topological relationships between them. Relationships captured by A&AI include "top to bottom" relationships such as those defined in ASDC when products are composed of services, and services are composed of resources. It also includes "side to side" relationships such as end to end connectivity of virtualized functions to form service chains. A&AI also keeps track of the span of control of each controller, and is queried by MSO and placement functions to identify which controller to invoke to perform a given operation.

A&AI is metadata driven, allowing new inventory item types to be added dynamically and quickly via AT&T Service Design & Creation (ASDC) catalog definitions, reducing the need for lengthy development cycles.

10.1 Key A&AI Requirements

The following list provides A&AI key requirements.

- Provide accurate and timely views of Resource, Service, and Product Inventory and their relationship to the customer's subscription.
- Deliver topologies and graphs.
- Maintain relationships to other key entities (e.g., location) as well as non-D2 inventory.
- Maintain the state of active, available and assigned inventory within ECOMP
- Allow introduction of new types of Resources, Services, and Products without a software development cycle (i.e., be metadata driven).
- Be easily accessible and consumable by internal and external clients.
- Provide functional APIs that expose invariant services and models to clients
- Provide highly available and reliable functions and APIs capable of operating as generic cloud workloads that can be placed arbitrarily within the AT&T AIC cloud infrastructure capable of supporting those workloads.
- Scale incrementally as ECOMP volumes and AIC Infrastructure scales.
- Perform to the requirements of clients, with quick response times and high throughput.
- Enable vendor product and technology swap-outs over time, e.g., migration to a new technology for data storage or migration to a new vendor for MSO (Master Service Orchestrator) or Controllers.
- Enable dynamic placement functions to determine which workloads are assigned to specific ECOMP components (i.e., Controllers or VNFs) for optimal performance and utilization efficiency.
- Identify the controllers to be used for any particular request.

10.2 A&AI Functionality

A&AI functionality includes Inventory and Topology Management, Administration, and Reporting & Notification.

Inventory and Topology Management

A&AI federates inventory using a central registry to create the global view of D2 inventory and topology. A&AI receives updates from the various inventory masters distributed throughout the D2 infrastructure, and persists just enough to maintain the global view. As transactions occur within D2, A&AI persists asset attributes and relationships into the federated view based on configurable metadata definitions for each activity that determine what is relevant to the A&AI inventory. A&AI provides standard APIs to enable queries from various clients regarding inventory and topology. Queries can be supported for a specific asset or a collection of assets. The A&AI global view of relationships is necessary for forming aggregate views of detailed inventory across the distributed master data sources within D2.

Administration

A&AI also performs a number of administrative functions. Given the model driven basis of ECOMP, metadata models for the various catalog items are stored, updated, applied and versioned dynamically as needed without taking the system down for maintenance. Given the distributed nature of A&AI as well as the relationships with other ECOMP components, audits are periodically run to assure that A&AI is in sync with the inventory masters such as controllers and MSO. Adapters allow A&AI to interoperate with non-D2 systems as well as 3rd party cloud providers via evolving cloud standards.

Reporting and Notification

Consistent with other ECOMP applications, A&AI produces canned and ad-hoc reports, integrates with the ECOMP dashboards, publishes notifications other ECOMP components can subscribe to, and performs

logging consistent with configurable framework constraints.

11. Business Support Systems Require Pivot to Take Advantage of D2

D2-based Offer/Products will be designed, created, deployed and managed in near real-time, rather than requiring software development cycles. ECOMP is the framework that provides service creation and operational management of D2 which enables significant reductions in the time and cost required to develop, deploy, operate and retire AT&Ts products, services and networks. The Business Support Systems which care for such capabilities as sales, ordering, and billing will interact with ECOMP in the D2 architecture, and those systems will need to pivot to this new paradigm.

While BSSs exist for today's network, they will need to change in order to work and integrate with ECOMP. These changes will need to be made with the assumption that the BSSs must also support existing products (perhaps in a new format) and enable Time to Market (TTM) new product creation & configuration on the D2 network.

The BSS transformation to support the dynamic D2 environment is based on the following:

Building blocks – BSS migration from monolithic systems to a platform building block architecture that can enable upstream User Experience (UX) changes in how products are sold, ordered, and billed.

Catalog Driven - BSSs will become catalog driven to enable agility, quick time to market and reduce Technology Development costs.

Improved data repositories – support accuracy and improved access to dynamically changing data (e.g., customer subscriptions).

Real-time APIs – BSS platforms must expose functionality via real-time APIs to improve flexibility and reduce cycle time.

New Usage Data & Network Events - BSSs that have a need to know about network events (e.g., billing) will be re-tooled to support new information from DCAE and A&AI.

Expose BSS functions – provide BSS functions directly to our customers to streamline processes and allow new distribution channels.

11.1 BSS Scope

The following Figure 13 shows the BSS Scope that includes Customer Management, Sales & Marketing, Order Lifecycle Management, Usage and Event Management, Billing, Customer Finance, User Experience, and End-to-End BSS Orchestration.

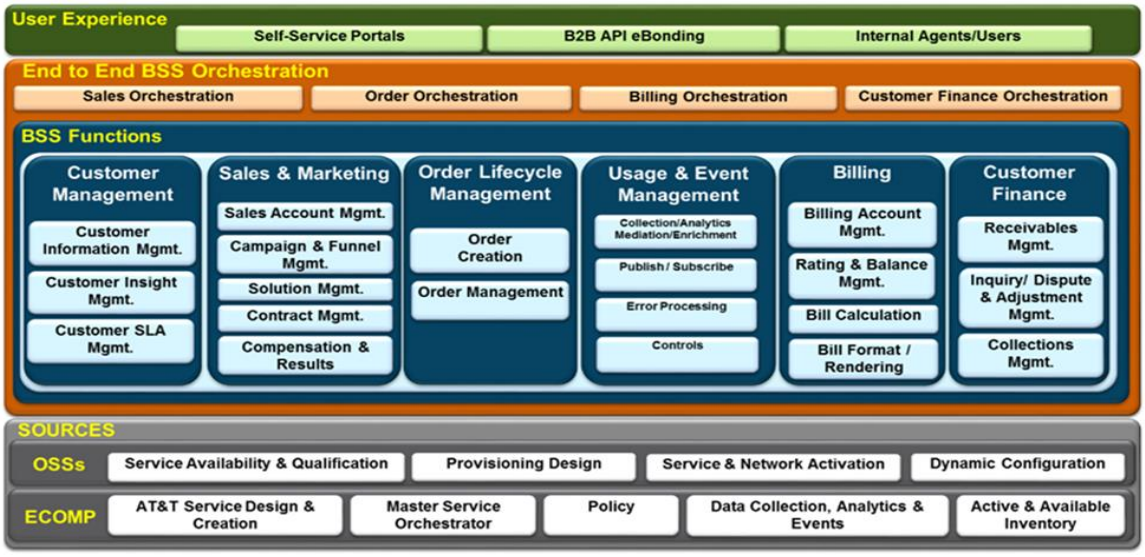

Figure 13: BSS Scope

The BSS scope includes the following areas.

Customer Management: focuses on customer information, retention of the customer, insight about the customer, managing customer service level agreements and building customer loyalty. Key new consolidated data stores of customer

information for D2 are the customer profile, customer subscription, and customer interaction history.

Sales & Marketing: provides all of the capabilities necessary to attract customers to the products and services AT&T offers, create solutions that meet customers' specific needs and contract to deliver specific services. Sales & Marketing hand off contracts to billing for implementation and solutions to ordering for service provisioning.

Order Lifecycle Management: provides the capabilities necessary to support the end-to-end handling of a customer's order. Orders may be for new service, or may be for moves, changes, or cancellation of an existing service. The experience of ordering will change in D2, as customers will experience provisioning in near real-time.

Usage and Event Management: focuses on end-to-end management of D2 usage and events, including transforming from the traditional vertically oriented architecture to a decomposed function based architecture. This will drive common collection, mediation, distribution, controls and error processing. The scope of this includes usage and events that are required for real-time rating and balance management, offline charging, configuration events, customer notifications, etc.

Billing: focuses on providing the necessary functionality to manage billing accounts, calculate charges, perform rating, as well as format and render bills. Billing will evolve under D2 to become more real-time, and decomposed into modules that can be individually accessed via configurable reusable APIs.

Customer Finance: manages the customer's financial activities. It includes the Accounts Receivable Management Functions, Credit & Collections Functions, Journals and Reporting Functions as well as bill inquiries, including billing disputes and any resulting adjustments. As in the case of Billing, Customer Finance will evolve under D2 to expose more API based components that are reusable across platforms.

User Experience: provides a single presentation platform to internal and external users each of which receives a customized view based on role. The user experience is divided into the self-service view for external customers, the B2B API Gateway for direct use of AT&T APIs by external applications and the internal view for AT&T sales agents, care center agents and other internal users.

End to End BSS Orchestration: These functions identify and manage the business level processes and events required to enable a customer request and interactions between domains. They trigger activities across domains and manage status, and provide a tool for Care to holistically manage the end to end request as well as the customer relationship for the duration of the enablement/ activation of the request.

11.2 BSS Interaction with ECOMP Components

BSS interaction with multiple ECOMP Components is critical to streamline the introduction of new offers, products and services.

AT&T Service Design and Creation - Offer/Product Creation

AT&T Service Design and Creation (ASDC) is key to the BSS agility we require for D2. ASDC Offer/Product models, recipes/templates and policies will be expressed as metadata that is distributed to BSS systems for real-time use or consumption in the execution of their work. BSSs will have to be significantly retooled to enable them to be configured by the information contained in ASDC. This will enable Customers to configure their own products via the self-service portal.

The ASDC platform contains the Master Reference Catalog which describes D2 assets in terms of a hierarchical Offer, Product, Service and Resource model. Only the Offer and Product Model are sent to the BSS layer. Products are created from Services by Product Designers and then Product Managers make products and offers sellable by adding descriptors containing pricing, discounting and promotion specifications. These design time relationships will be through the Offer and Product levels of the ASDC catalog. Each offer and product in the Model has a description (profile) with a recipe (model) containing processes, policies & rules that is distributed and stored locally in each BSS.

Each ASDC level has associated definitions, processes, and policies for management and execution. BSSs will integrate seamlessly with ECOMP via these shared definitions and models to radically improve time-to-market for new services, products, and offers. This will also enable Customers to configure their own products via the "Customer Product Composer" UI by

chaining products together from the same product definitions as the Product Designer would use. These interactions will be through the ASDC catalog Offer and Product level. ASDC distributes Offer and Product models to BSSs, and distributes Service and Resource models to ECOMP components.

MSO (Master Service Orchestrator)

The Master Service Orchestrator's (MSO's) primary function is the automation of end-to-end service instance provisioning activities. MSO provides the BSS systems an interface to orchestrate delivery of D2 services.

BSS End to End Orchestration will decompose the customer request in to D2 and non-D2 services based on rules from the product catalog. The BSS End to End Orchestration triggers the MSO to initiate ECOMP activity, and the MSO manages the provisioning/network activity by interacting with Controllers as required. The MSO sends the business level status back to the BSS Orchestration. The BSS Orchestration does not maintain provisioning logic nor manage provisioning processes.

Data Collection, Analytics and Events (DCAE)

DCAE provides the infrastructure for collection of autonomous events from the network and other D2 components, making them available to subscribing applications, including the BSS applications.

The DCAE environment forwards usage and other information that can be used by the BSS to generate billable events, Usage Event Management for collecting D2 usage, and other events and records. Based on the timing requirements provided by Usage Management and other applications (e.g., seconds, minutes vs. hours vs. days), the BSSs will obtain the data from DCAE distribution channels or from the Data Lake. For example, the Billing function in a BSS can support near real-time balance management by receiving streaming analytics from DCAE.

Usage and Event Management BSSs can be created as applications on top of the DCAE environment as well as applications outside DCAE. The allocation of these applications will be determined as use cases are further developed and the associated allocation of functions are delineated. Usage and Event Management applications will collect customer events and perform mediation of the usage/events to downstream BSSs/OSSs. Similarly, BSSs can collect network events such as bill impacting configuration changes, consumption or any new bill impacting network product or service. BSSs use the data for rating, balance management, charge calculations, etc.

ECOMP Data sources will provide data to BSS for Customer SLA credit processing. SLA Data Sources will collect information needed to perform Customer SLA Management. The ECOMP Data Sources are as follows: DCAE – network measurement, ECOMP Orchestration – service orchestration (activation), Policy – rules/violations, ASDC – SLA Definition, data templates for products. Note that both ECOMP Data sources and BSS will need to be informed about the fields to be gathered for those SLAs by the product definition in ASDC. The actual determination if an SLA has been violated, calculation of credits, and applying those credits to the customer bill will be done by BSS, not ECOMP Data sources. However, if there is an elastic response required based on a SLA violation being approached, the service may require an action by ECOMP to monitor and respond to a defined policy.

Active & Available Inventory (A&AI)

Customer Information Management (CIM) / Customer Subscription Management maintains a view of items purchased or in the process of being purchased by a customer. Customer Subscription entitlements are created by service designers in advance then bound to Product/Service/Resource inventory instances in ECOMP's Active & Available Inventory (A&AI) at the time they are provisioned. Non-virtual items may still be related to one another in static infrequently changing relationships. Customer Subscription will be able to associate the product instance to the subscription instance which will enable support of DCAE functions such as routing events to the BSS about the customer subscription. Subscription data contains linkages to the AT&T Service Design & Creation (ASDC) platform's Product Catalog for static attributes (e.g., product name and type) and pricing and adjustment lists. Subscription items will also contain certain configurable parameters that may be changed by the customer.

Data collection performed by the Usage and Event Management applications built on top of the DCAE will require interaction with A&AI. This includes association of all events belonging to a given Product Instance ID. The A&AI view will be used to aggregate all of the events from resource instance to service instance to product instance. These BSS applications

will then associate the Product Instance ID to the Customer via the corresponding Customer Product Subscription.

Policy

BSSs interact with Policy at the Product layer when Product Designers set the initial Policy associated with a Product in the Product level of the ASDC Catalog. This can take the form of minimum Service Level Agreement guaranteeing certain levels of uptime, response and throughput in traditional products or set as ranges with associated price levels and left open for the customer to specify later.

The customer purchases a product which creates an order containing customer subscription information including the initial product level parameters which get passed to ECOMP. These are used by ECOMP to set up the customer service with underlying service level policies. The customer can then optimize their network performance within a certain product allowed minimum and maximum range through the customer portal by setting these parameters themselves, e.g., varying Class of Service or Bandwidth. Other limitations set by policy could be maximum allowed changes per day, minimum time period between changes, etc.

While Policy can be created and employed by DCAE at any layer of the execution environment, BSS sets Policy at the Offer/Product/Service level visible to the customer. At the Product level these would be rules that govern the workings of an AT&T Product for a particular customer at a point in time. Policy usage in ECOMP focuses on closed loop patterns that will keep the Product performing at the Customer's desired level. BSSs will move from using tightly bound service Policy parameters coded into each application as part of a Technology Development release to dynamically driven by ASDC.

12. Security

There are two main aspects to security in relation to the ECOMP Platform: security of the platform itself and the capability to integrate security into the cloud services. These cloud services are created and orchestrated by the ECOMP platform. This approach is referred to as security by design.

The enabler for these capabilities within ECOMP is an API-based Security Framework, depicted in Figure 14 below. This illustration also shows how the AT&T security platforms work with ECOMP to provide security functionality. The AT&T Security platform is connected to ECOMP through a set of security APIs. While the security platform is specific to AT&T, the ECOMP framework can be utilized by other security platforms. This framework allows for security platforms and applications existing outside of ECOMP to be used for platform security and security by design for services it orchestrates.

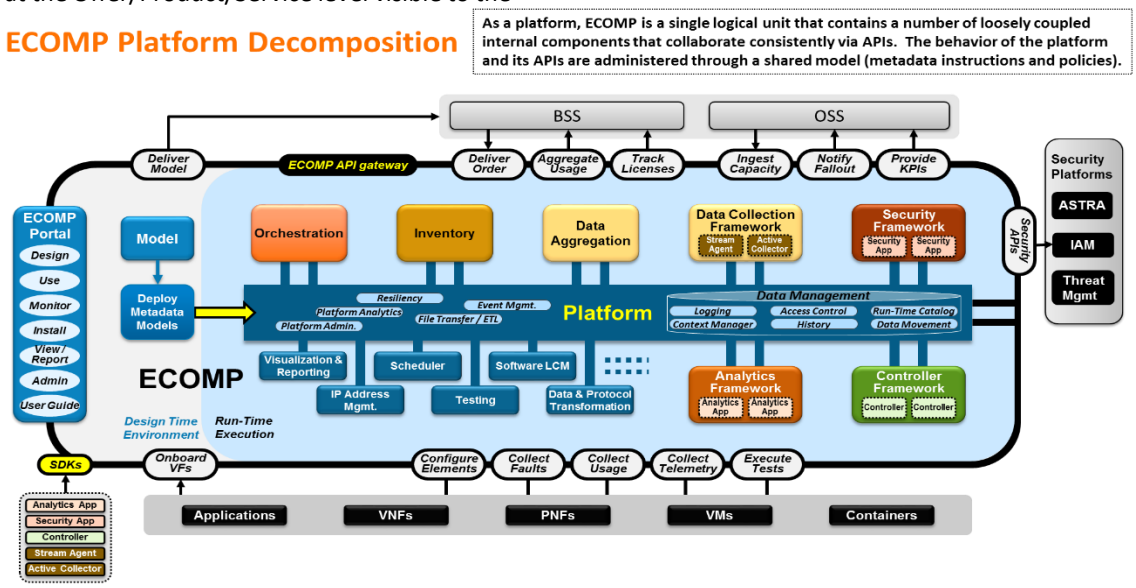

Figure 14: ECOMP Platform Decomposition

Security of the platform begins with a strong foundation of security requirements and following security best practices as an inherent part of the ECOMP design. Some examples include:

- deployment of the platform on a secure physical and network infrastructure
- adherence to secure coding best practices
- security analysis of source code
- vulnerability scanning
- defined vulnerability patching process

Building upon this foundation, external security platforms that provide additional security capabilities such as identity and access management, micro-perimeter controls and security event analysis are integrated onto the platform through advantageous use of the ECOMP Security Framework. The additional security these external platforms provide are described below.

Security modules such as the Identity and Access Management (IAM) platform provide critical security capabilities to the ECOMP solution. Access management enhancements deliver preventive and detective access controls for the ECOMP portal and related front ends. Options for fine grained authorization capability also exist. For identity lifecycle management, this platform provides user provisioning, access request, approval and review capabilities and is designed to minimize administrative burden.

Internal to AT&T, security such as micro-perimeter controls can be provided by Astra, the AT&T-developed innovative and award winning[1] cloud security platform; this platform enables continuous protection for the AT&T Integrated Cloud (AIC). The Astra security ecosystem and framework allows virtual security protections to be enabled effortlessly via APIs and automated intelligent provisioning, creating micro-perimeters around the platform and applications. Astra enables security function virtualization as well as dynamic real-time security controls in response to the ever evolving threat landscape. For example, based on security analytics using big data intelligence, Astra enables virtual security functions on-demand, leveraging our SDN enabled network, to dynamically mitigate security threats.

Security event analysis, provided by a security analytics platform, will use the ECOMP DCAE data collection and analytics engine to gather VNF data, network data, logs and events. Once the security analysis has determined that a security event has occurred, a pre-determined policy can be invoked via the ECOMP platform. The ability to respond automatically to a security-related event, such as a Distributed Denial of Service (DDoS) attack, will enable closed loop security controls, such as modifying firewall rules, or updating Intrusion Prevention System (IPS) signatures, etc. In the event that a pre-determined policy has not been created for an event, it will be sent to a ticket system, and then a new policy can be generated for the next time that event occurs.

The ECOMP platform also enables security by design for services it orchestrates by engaging a security trust model and engine. This begins with validation of security characteristics of resources as part of the ASDC resource certification process. This assures service designers are using resource modules that have accounted for security. Using the ECOMP security framework to access an external security engine, additional security logic can be applied and enforced during service creation.

ECOMP is a platform for many types of services. Because of its inherent security, it is also a powerful means to provide security as a service. In many ways, security services are similar to other services; however, even more so than other services, security services must be provided via a platform / infrastructure that is inherently secure.

Many types of security services can be offered, spanning access control, authentication, authorization, compliance monitoring, logging, threat analysis and management, etc. Management of vFW (virtual Firewall) capabilities can be described to illustrate this opportunity. For example, when a customer has a need for firewall capability, the customer provides the needed information via the portal to enable ECOMP to determine and orchestrate the firewall placement. In addition, the firewall capabilities (e.g., rules, layer 7 firewall) are instantiated at the appropriate locations within the architecture. If necessary, many security controls and technologies including firewalls, URL blocking, etc., can be service-chained to provide all the needed functionality. As part of an overall security architecture, the log data from the firewalls can be captured by DCAE and used by the threat management application to perform security

[1] ISE® Northeast Project Award Winner 2015

analytics. Should a threat be detected, various mitigation steps can be taken, such as altering IPS settings, change routing, or deploy more resources to better absorb an attack. This can be achieved by Astra working with ECOMP to deploy the appropriate updates across the infrastructure, thereby minimizing the service interruption due to the security threat.

13. Today and Tomorrow

In 2015, AT&T successfully deployed 74 AIC nodes (exceeding its target of 69) and surpassed its goal of virtualizing 5% of the target network load. These achievements in virtualization were largely made possible by the delivery of various ECOMP Platform components in support of early D2 projects such as Network Function on Demand (NFoD), Hosted BVoIP platform consolidation, Mobility Call Recording (MCR), Virtual Universal Service Platforms (vUSP) and IP Interconnect. Initial ECOMP Platform capabilities were released at a component level, with the operational components of A&AI, Infrastructure Controller, MSO, Network Controller, DCAE, and Portal seeing multiple releases. The design components of ASDC and Policy had initial releases centered upon basic meta-data driven capabilities, with a plan to rapidly build out the more sophisticated modeling framework, including complex policy creation, distribution and enforcement.

Much of ECOMP's success can be attributed to the combined use of agile development methodologies and a holistic architecture approach. The end state consists of multiple policy driven control loops with template/pattern/recipe driven application flows resulting in a dynamically controlled D2 environment. This environment cannot be directly managed by human operators and requires the support of intelligent automation constructed using a DevOps approach which synergizes the combined expertise of software experts, network experts, and operations SMEs. Incremental agile delivery and DevOps are transforming AT&T's technical culture in lock step with overall D2 evolution and are corporately recognized as key success factors.

In the near future, ECOMP will be providing open platform capabilities via the ECOMP Design Framework that enables 3rd parties to make use of, integrate with, create and enhance capabilities of D2 functions or services. Key functions will be exposed via open APIs which align to industry/AT&T standards and are supported by an open and extensible information/data model. A success factor is that applications/services and application/service components are programmable by AT&T (and users in many cases) via policy and rules to eliminate/minimize the need for per service developments. Services are expected to move through a lightweight development process where completion of the software effort is in an AIC "sandbox" environment which can be directly toggled into production, ready for fully automated deployment and lifecycle management.

To expedite the delivery of Platform components, AT&T has a preference towards open source software and industry standards. AT&T will engage with D2 Suppliers for co-development and off-the-shelf software when appropriate.

14. Summary

ECOMP is an open software platform, capable of supporting any business domain. It is designed and built for real-time workloads at carrier scale. It is currently optimized to deliver an open management platform for defining, operating and managing products and service based upon virtualized network and infrastructure resources and software applications. As an open software platform, it includes components enabling a reduced time to market via the rapid introduction of new functions using dynamic definitions and implementations based on standard metadata and policies all managed using a visual design studio. As a network management platform, it includes a framework consistent across cloud infrastructure, applications and network management components, that enables rapid introduction of compute, memory, and storage resources used to dynamically instantiate, operate and lifecycle manage services, virtualized network functions, applications and smart cloud infrastructure. The network management platform generates value by enabling the virtualization of network functions via automation of the definition, delivery and management of virtualized functions, and by the dynamic shaping and placement of infrastructure-agnostic workloads. Value is further enhanced by enabling interoperability across 3rd party cloud infrastructures and between virtualized networks.

ECOMP is critical in achieving AT&T's D2 imperatives: increase the value of our network to customers by rapidly on-boarding of new services (created by AT&T or 3rd parties), reduce CapEx and OpEx, and provide Operations efficiencies. It delivers enhanced

customer experience by allowing them in near real-time to reconfigure their network, services, and capacity. ECOMP enables network agility, elasticity, and improves Time-to-Market/Revenue/Scale via ASDC visual modeling and design. ASDC utilizes catalog-driven visual modeling and design that allows the quick on-boarding of new services, reducing cycle time from many months to a few days, and facilitates new business models and associated monetization paradigms. ECOMP provides a Policy-driven operational management framework for security, performance and reliability/resiliency utilizing a metadata-driven repeating pattern at each layer in the architecture. This approach dramatically improves reliability and resiliency as well as operational flexibility and speed. ECOMP reduces CapEx through a closed loop automation approach that provides dynamic capacity and consistent failure management when and where it is needed. This closed loop automation approach reduces capital cost for spare capacity deployed today for worst case failover scenarios. Managing shared resources across various network and service functions is enabled by aggregating capacity thus maximizing overall CapEx utilization. ECOMP facilitates OpEx efficiency though the real-time automation of service/network delivery and lifecycle management provided by the OMF framework and application components. The nexus of these capabilities is a family of deterministic control loops driven by ASDC and Policy recipes that eliminates many of the manual and long running processes performed via traditional OSS's (e.g., break-fix largely moves to plan and build function). AT&T achieves economy of scale by using ECOMP as a single platform that manages a shared AIC infrastructure and provides operational efficiency by focusing Network and Service Management control, automation and visualization capabilities on managing scale and virtualized application performance and availability.

The ECOMP platform provides external applications (OSS/BSS, customer apps, and 3rd party integration) with a secured, RESTful API access control to ECOMP services, events, and data via AT&T gateways. In the near future, ECOMP will be available in AT&T's D2 Incubation & Certification Environment (ICE) making ECOMP APIs available to allow vendors, cloud providers, and other 3rd parties to develop solutions using ECOMP and AIC reference architecture (current and future-looking).

For Further Information

To provide technical feedback on the whitepaper, or express interest in driving this initiative forward, write us at ecomp-feedback@research.att.com. For ECOMP supply chain *questions* please contact https://www.attsuppliers.com/domain2.asp.