Autonomous Services Composition in Domain 2

Mazin Gilbert, Anwar Syed Aftab, Farheen Cefalu, Pamela Dragosh,
Rittwik Jana, Serban Jora, Thomas Kirk, John Lucas, Arthur Martella,
John Murray, Sundar Ramalingam, Christopher A Rath, Shu Shi, Rich Wright, Avi Zahavi

AT&T is embarking on an exciting journey to revolutionize its network by transforming itself into a software company running the largest and most intelligent programmable cloud on the planet. Indeed, the network of Domain 2.0 (D2) will be intelligent software systems and applications operating on general-purpose commodity hardware [1]. This transformation will not only drive down CAPEX, OPEX and help to configure our network with less human intervention, but it will also create significant opportunities to scale and monetize existing and new intelligent services. This transformation will enable AT&T's D2 to establish a new services ecosystem equivalent in concept to the application ecosystem adopted by the Apple iOS and Android. D2 will facilitate mass marketing existing and new services, and lower the barrier to entry for enterprise and small business customers to create new innovative services.

This paper will provide a demonstration of how intelligent services can be realized within the D2 architecture. An intelligent service collects information about resources (users, devices, networks and applications) used to deliver the service and about the environment in which it operates. It then makes decisions based on this information and domain knowledge which includes adapting and personalizing the service for the users consuming it. An intelligent service receives feedback on its performance and learns. There are primarily three attributes that characterize an intelligent service, known as PAD (personalized, adaptive, and dynamic). A predictive personalized service is one that anticipates a user's need and proactively takes intelligent actions and recommends valuable and timely personalized information. An adaptive service learns from its past actions of all users and adjusts its behavior to provide superior service quality. A service that is dynamic is one that is robust and is able to survive and self-repair or self-organize itself from possible service interruptions.

In this demo paper, Autonomous Services Composition (ASC) is one example of an intelligent service (see Figure 1). ASC's vision is to create a `services marketplace' that provides a holistic customer experience to create novel services by leveraging advanced tools like recommenders and expert systems, real time customer care collaboration and big data analytics. Service composition is a mechanism for integrating applications/services from predefined component modules (e.g. resources) and deploying those modules where and when they are needed. It is the service linking capability which provides the basic mechanism for defining new applications, managing application lifecycles and implementing elastic services in the D2 environment. By rapidly integrating components and sizing those components dynamically, applications can be built to handle varying user loads. By selectively deploying component modules based upon affinity rules, multiple time-critical functions can be directed to share common infrastructure to minimize latency, while high availability can be maintained by directing redundant components to execute on geographically separated servers.

Example: Consider the following set of basic services: a) data connectivity between two endpoints (e.g. switched Ethernet service like Gamma), b) a data `splitter' service that takes data from an endpoint and sends it to two or more endpoints, c) a video anomaly detection service that takes a video stream and sends alarms to an endpoint, and d) a data archiving service. A small-town bank with a video surveillance camera in the lobby wants to send that video stream to a home office while away from the bank. In ASC, we configure the basic connectivity service with the endpoint locations. Alternatively, alerts can be sent when something unexpected happens. In ASC, we can compose by taking the video output from the data splitter service and piping it into the anomaly detection service, and configure it to send alerts out. There may be a need to store the video for six months as well as receiving the alerts. In all of these cases, ASC is creating the individual service orders for the basic services, and the overall application control that shows how they are stitched together.

Challenges and problems solved

There are many challenges that service composition presents.

NFV modeling: ASC uses items from the resource, service and product catalogs. There is a need to model these Network Function Virtualization (NFV) elements before composition can be realized. We have extended Open Topology and Orchestration Specification for Cloud Applications (TOSCA) to model these products. This facilitates rapid service composition in a manner that is verifiable at different stages of the software lifecycle. The starting point of composition is a catalog of third party NFVs. We are creating an ecosystem that will bootstrap our vendors to populate the resource and service catalogs using TOSCA.

ASC system architecture: The architectural approach we have adopted for the ASC system enables rapid iterative development, an elastic scaling path as utilization grows, and flexibility in terms of evolving requirements and addition of new functionality. The subsystems of ASC are designed around a *microservice architecture*, in which components interoperate via a lightweight distributed message bus (Vert.x). The carefully abstracted messaging pub/sub interface facilitates extensibility (e.g. as we add semantic assistants from the intelligent toolbox). Finally, the cloud-based ASC cluster can easily scale horizontally to meet elastic demand -- subsystem replicas can be deployed in seconds on lightweight containers as utilization requires.

Composition is ultimately a complex data-driven process, using not only an extensive product catalog and asset inventory, but also representations of the domain expertise that goes into building complex composite services. We're using graph database technology to allow natural ways to represent the semantic connections among catalog and inventory items, workflow state, and related domain knowledge. The database is dynamically extensible to accommodate learning throughout the lifecycle, and is continuously available to intelligent agents overseeing and augmenting the composition process.

Integration with D2: ASC needs to finally execute the composed order in the D2 environment. A `shopping cart' like experience has been created that provides the end customer to discover, create, deploy and watch the service. ASC needs to interoperate with the Service Design and Creation (SD&C) subsystem to fulfil the order. ASC also communicates with a range of D2 entities (e.g. DCAE engine in ECOMP, A&AI for real-time asset information) to monitor the service and present real-time reports in a dashboard using analytics and visualization tools.

Expert systems and collaboration: A natural part of the composition process is to allow the customer to discover services and products that are related. An expert system is needed to suggest or recommend compatible products to guide successful compositions. Another key feature of ASC is designing for collaboration; we know that composition is a complex activity often involving many participants, so we have designed the ASC environment to be a collaborative workspace from the start. Another aspect of composition of AT&T services is the complexity and richness of the workflow, so the ASC environment seeks to provide seamless support of the entire composition lifecycle. To make the ASC environment maximally accessible, the user interface is entirely browser-based, leveraging HTML5 technologies like WebGL and WebRTC to enable a powerful and effective user experience. Finally, we have built a web RTC based collaboration environment for customer care needs.

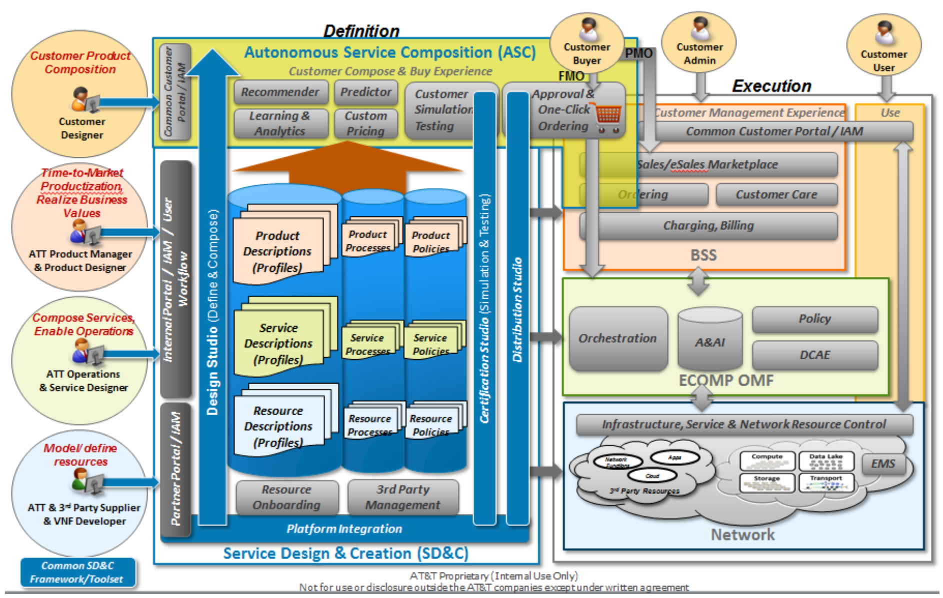

Figure 1 - Autonomous Services Composition Target Architecture

References

[1] Chiosi M., "AT&T Domain 2.0 Vision White Paper", http://www.att.com/Common/about_us/pdf/AT&T%20Domain%202.0%20Vision%20White%20Paper.pdf, Nov. 13, 2013.