International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

J.1302

(06/2021)

SERIES J: CABLE NETWORKS AND TRANSMISSION OF TELEVISION, SOUND PROGRAMME AND OTHER MULTIMEDIA SIGNALS

Cloud-based converged media services for IP and broadcast cable television

Specification of a cloud-based converged media service to support Internet protocol and broadcast cable television – System architecture

Recommendation ITU-T J.1302

Recommendation ITU-T J.1302

Specification of a cloud-based converged media service to support Internet protocol and broadcast cable television – System architecture

Summary

Recommendation ITU-T J.1302 defines the high-level system architecture of a cloud-based converged media service to support Internet protocol (IP) and broadcast cable television (TV). With the cloud-native technology development, cloud-based converged media services can be quickly deployed by cable television operators. This Recommendation is part 2 of a multi-part deliverable.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T J.1302	2021-06-13	9	11.1002/1000/14647

Keywords

CBCMS, system architecture, microservice.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity, or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Introduction

This Recommendation is part 2 of a multi-part deliverable covering the system architecture for cloud-based converged media service to support Internet protocol and broadcast cable television, as identified below:

Part 1: Requirements.

Part 2: System architecture.

Part 3: System specification on collaboration between production media cloud and cable service cloud.

Recommendation ITU-T J.1302

Specification of a cloud-based converged media service to support Internet protocol and broadcast cable television – System architecture

1 Scope

This Recommendation specifies the system architecture for cloud-based converged media services (CBCMSs) to support Internet protocol (IP) and broadcast cable television (TV). The CBCMS is intended to be deployed in the service cloud of cable television operators, which enables rapid innovation and deployment of new services, flexible expansion of online services, smart adaption of service origins or paths to meet the distribution needs for different terminal types and network status. This system architecture fulfils the requirements in [ITU-T J.1301].

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T J.1301] Recommendation ITU-T J.1301 (2021), *Specification of a cloud-based converged media service to support Internet protocol and broadcast cable television – Requirements*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 integrated broadcast and broadband (IBB) DTV service [b-ITU-T J.205]: A service that simultaneously provides an integrated experience of broadcasting and interactivity relating to media content, data and applications from multiple sources, where the interactivity is sometimes associated with broadcasting programmes.

3.1.2 cloud-based converged media service (CBCMS) [ITU-T J.1301]: The cloud-based converged media service is intended to be deployed by cable television operators and to support the requirements of rapid service innovation and deployment, which enables media services to be developed by separate vendors according to the standard application programming interfaces (API), where appropriate.

3.1.3 microservice [ITU-T J.1301]: An architectural and organizational approach to software development where software is composed of small independent microservices that communicate over well-defined APIs. Microservices architectures make applications easier to scale and faster to develop, enabling innovation and accelerating time-to-market for new features.

3.1.4 cloud digital video recorder (DVR) [ITU-T J.1301]: Cloud DVR saves TV shows in the service provider's cloud datacentre rather than in the DVR/set-top box by the TV. Cloud DVRs enable customers to store more content and record more shows that air at the same time.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 API gateway: In microservices architecture, applications and services are composed of smaller, exchangeable components, and these components need a way to find and communicate with one another. This is where API gateways come in. An API gateway sits between clients and services. It acts as a reverse proxy, routing requests from clients to services. It may also perform various cross-cutting tasks such as authentication, transport layer security (TLS) termination, and rate limiting.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AC	Access Criteria
Ad	Advertisement
AI	Artificial Intelligence
API	Application Programming Interface
CA	Conditional Access
CAS	Conditional Access System
CBCMS	Cloud-Based Converged Media Service
CDN	Content Distribution Network
CW	Control Word
DASH	Dynamic Adaptive Streaming over HTTP
DRM	Digital Rights Management
DVR	Digital Video Recorder
DVB	Digital Video Broadcasting
ECM	Entitlement Control Message
ECMG	Entitlement Control Message Generator
EMMG	Entitlement Management Message Generator
HDS	HTTP dynamic streaming
HLS	HTTP Live Streaming
HSS	HTTP Smooth Streaming
ID	Identification
SQL	Structured Query Language
TLS	Transport Layer Security
UI	User Interface
UX	User Experience
VOD	Video On Demand
VM	Virtual Machine

5 Conventions

In this Recommendation:

The keywords "**is required to**" indicate a requirement that must be strictly followed and from which no deviation is permitted if conformance to this Recommendation is to be claimed.

The keywords "**is recommended**" indicate a requirement that is recommended but which is not required. Thus, this requirement need not be present to claim conformance.

The keywords "**is prohibited from**" indicate a requirement that must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

The keywords "**can optionally**" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

In the body of this Recommendation and its annexes, the words *shall*, *shall not*, *should*, and *may* sometimes appear, in which case they are to be interpreted, respectively, as *is required to*, *is prohibited from*, *is recommended*, and *can optionally*. The appearance of such phrases or keywords in an appendix or a material explicitly marked as *informative* is to be interpreted as having no normative intent.

6 High-level system architecture

Cloud-based converged media service (CBCMS) is the cloud-based media service that provides various capabilities according to the practical development requirements [ITU-T J.1301]. CBCMS employs a hierarchical and modular software architecture and consists of four functional layers: the infrastructure layer, the ability layer, the abstraction layer, the application layer, as well as management tools that apply to every layer. Each consists of multiple software modules in a loose-coupling mode.

Figure 1 shows the high-level system architecture of CBCMS.

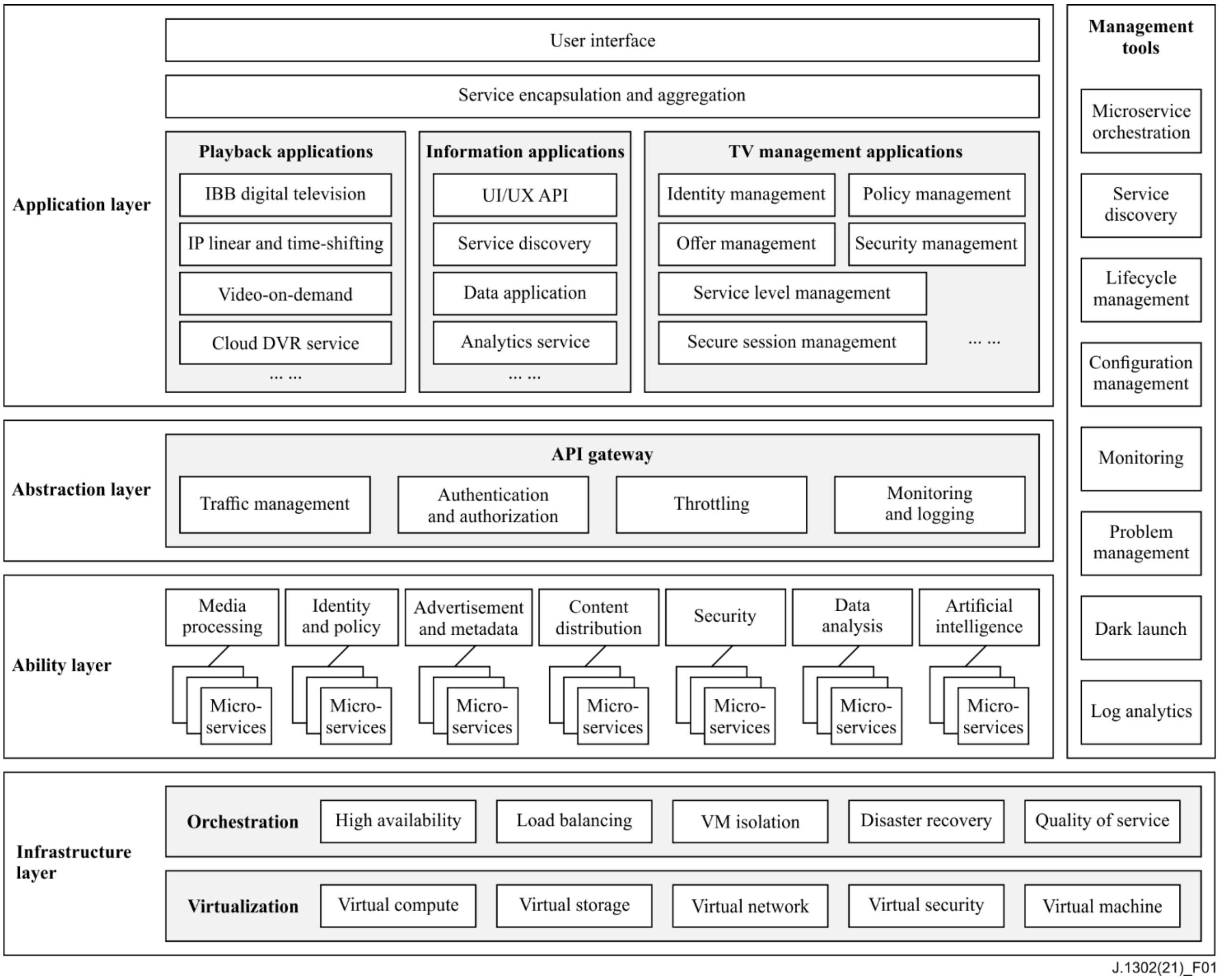

Figure 1 – CBCMS high-level system architecture

The infrastructure layer provides dynamically scalable compute, storage, network, and security resources based on virtualized infrastructure. It enables multiple applications to transparently share common underlying physical resources. The infrastructure layer provides basic resources for the upper-layer software.

The ability layer provides core abilities for converged media services. It includes the capabilities of media processing, identity and policy, operation, content distribution, security, data analysis and artificial intelligence.

The abstraction layer provides the abstraction functions including traffic management, authentication and authorization, throttling, monitoring, and logging. It exposes reusable services to the application layer.

The application layer consists of user interface, service encapsulation and aggregation, playback applications, information applications and TV management applications.

The management tools are composed of a series of components, which are required to register and orchestrate microservices, manage basic functions, and other components' managements for the system architecture.

7 Infrastructure layer

The infrastructure layer consists of the virtualization layer and orchestration layer. The virtualization layer provides virtual compute, virtual storage, virtual network, and virtual security resources based on the hardware platform. The orchestration layer provides high availability, load balancing, virtual machine isolation, disaster recovery, and quality of service, which can be quickly expanded or decreased to meet changing demands.

The infrastructure layer is required to extend high availability to applications by using techniques such as an automated virtual machine (VM) restart in conjunction with network storage. It is recommended to provide disaster recovery by rollover designated applications to another site within a resource pool.

It is recommended to take advantage of continuing technology improvements to further increase security and quality of service for the upper applications running in the shared multi-tenant environment.

8 Ability layer

The ability layer provides the core capabilities that can be reused by upper applications to provide full functionality within the framework. These capabilities must be developed in a way that meets core cloud architecture principles of fast velocity, automatic testing and deployments, enabling separate technology and deployment strategy per microservice.

The ability layer adopts a microservice architecture and defines seven types of capabilities, including media processing capability, identity and policy capability, advertisement and metadata capability, content distribution capability, security capability, data analysis capability, and artificial intelligence capability. Different combinations and instantiations of the ability layer and its microservices can support the converged media services.

The ability layer microservices are shown in Figure 2. This list is not exhaustive and other microservices could be added to the ability layer.

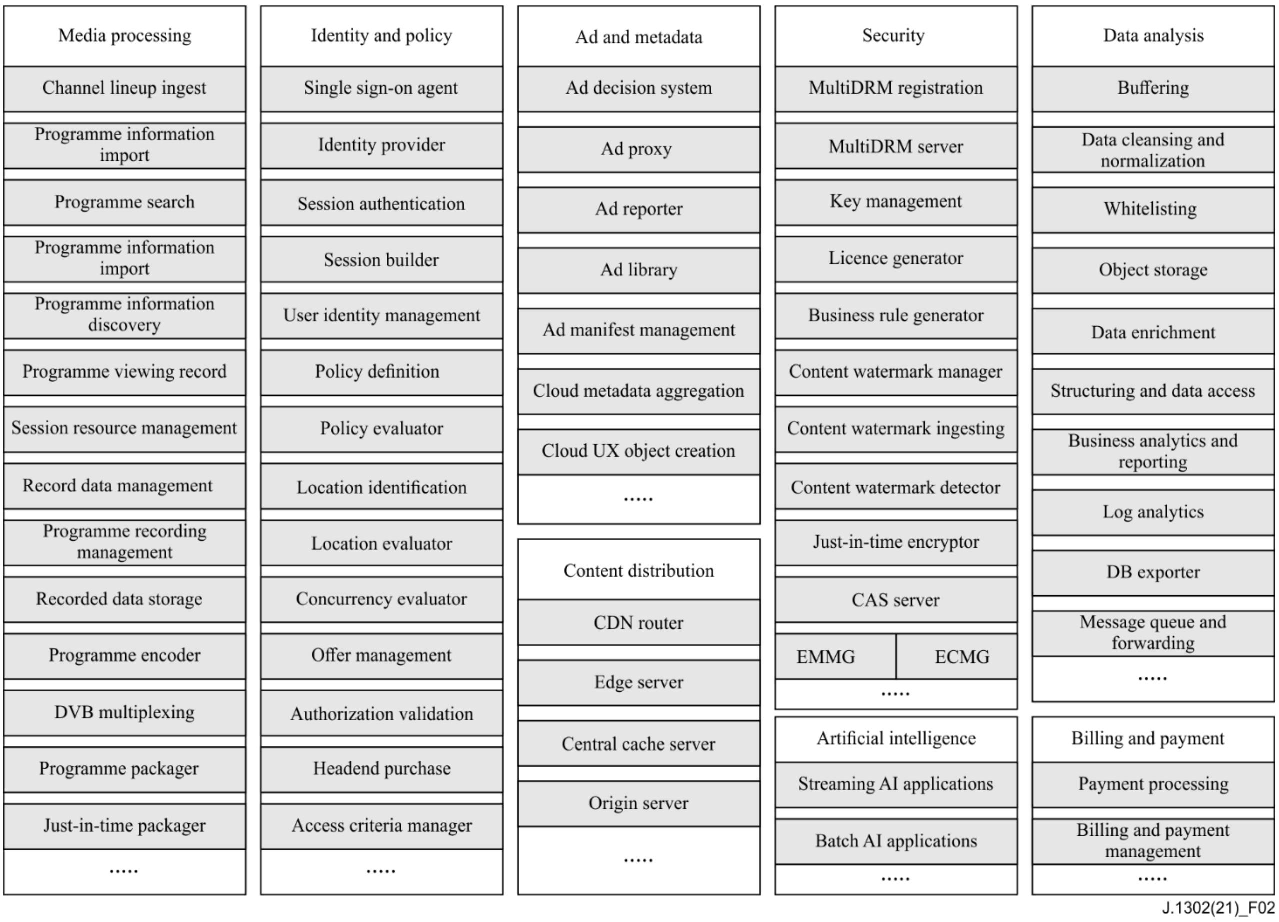

Figure 2 – The ability layer

8.1 Media processing capability

The media processing capability mainly supports video services such as liner, video on demand (VOD), DVR, etc., which includes channel lineup ingest, programme information import, programme event search, programme event recommendation, programme information discovery, programme viewing record, session resource management, programme encoder, digital video broadcasting (DVB) multiplexing, programme just-in-time packager, programme recording management, record data management and recorded data storage.

1) Channel lineup ingest

Microservices responsible for ingesting channel lineup, order, properties, and bouquets across all channels, where the properties could be the followings: channel name, channel number, bouquet Id, the channel order in bouquet, language, radio frequency (RF) information (for DVB service), channel genre, region, isAdult, isOTT, isDTH, etc.

2) Programme information import

Microservices responsible for ingesting schedules of linear or VOD programmes, including lists of contiguous events, content information, event information, posters, and other multi-media metadata. Regarding VOD programmes, the VOD content organization structure, VOD content and relevant properties should be imported as well, where properties may have: asset identification (ID), VOD title, VOD language, provider ID, asset duration, parental rating of asset, synopsis, actors, directors, keywords, production year, etc.

3) Programme search

Microservices include search capability to locate live programmes, time-shifting programmes, and VOD assets included in the VOD library.

4) Programme recommendation

Microservices that make personal recommendations of upcoming linear events or VOD assets to a particular user or account based on past consumption habits.

5) Programme information discovery

Microservices that provide linear schedule or VOD content organization in a specific file format to other microservices for processing (for example, to provide to user experience (UX) service to generate user interfaces).

6) Programme viewing record

Microservices responsible for maintaining the history of all programmes watched per account and device.

7) Session resource management

Microservices that validate authorization, parental rating, geolocation, and concurrency restrictions, etc. with offer management and availability with the policy management. Once authorized and in-policy, this microservice provides all necessary information to the client to enable consumption of contents, such as linear, VOD, time-shift TV (TSTV), DVR.

8) Record data management

Microservices that configure and manage specific service provider-approved live events for time-shifting or recording. They support both user interface (UI) discovery of time-shift content via linear UI and schedule a recording with the programme recording management microservice the data plane.

9) Programme recording management

Microservices that record programme content to enable time-shifting or DVR services.

10) Recorded data storage

The microservices contain all the recorded content for replay by the user, for time-shifting and DVR services.

11) Programme encoder

Microservices that encode baseband video into a compressed format such as MPEG4, HEVC, or AVS etc. to support the content delivery and consumption by the user device.

12) DVB multiplexing

Microservices that multiplex encoded programmes into DVB stream, and can also perform programme scrambling when needed, and send the scramble control word (CW) to the conditional access system (CAS) secure microservice.

13) Programme packager

Microservices that package content into a neutral format (CIF) for later instantiation and encryption by just-in-time packager.

14) Just-in-time packager

Microservices that package the linear, time-shifting, VOD and DVR contents based on the player type and required format such as HTTP live streaming, HTTP dynamic streaming, HTTP smooth streaming, and dynamic adaptive streaming over HTTP, (HLS, HDS, HSS or DASH). When content protection is required, they will encrypt the encoded content. The microservices are used only for delivering content in IP transportation. When used for advertisement (Ad) service, it will package the selected Ad into the format consistent with the player and provides Ad stitching with Ad manifest file.

8.2 Identity and policy capability

The identity and policy capability provides identity and policy microservices for the converged media services, to implement functions such as identity management, session management, and authorization management. The microservices are single sign-on agent, session authentication, session builder, user identity management, policy definition, policy evaluator, location identification, location evaluator, concurrency evaluator, etc.

1) Single sign-on agent

Microservice provides single sign-on with identity provider service, issues single sign-on access token and refresh tokens, registers account and new device properties.

2) Identity provider

Microservice that authenticates initial login of new device credentials to ensure that credentials are recognized as part of a registered account. It manages periodic reset of credentials.

3) Session authentication

Initiates the creation of TV viewing sessions using session cookies based on validation of the access token. It is required to validate session cookie before enabling access to over-the-top (OTT) media service.

4) Session builder

To build TV viewing sessions using the session cookie method. TV viewing session is a generic terminology, which includes all kinds of sessions related to TV viewing, such as linear, VOD, DVR, etc.

5) User identity management

Manages all devices, users, offers and policies in each account. Enables setting and modification of these properties by role-based authorized services. Notifies other services of these changes.

6) Policy definition

Maintains the definition of all policies across linear with time-shifting, VOD, and cloud DVR services.

7) Policy evaluator

Evaluates whether a policy is being maintained by every user, device, or account.

8) Location identification

Identifies location of client devices based on usage of location mapping with IP address.

9) Location evaluator

Evaluates whether location or proximity restrictions defined in the policy definition are being met; it works with all TV services to help session resource management.

10) Concurrency evaluator

Evaluates whether concurrency restrictions defined in the policy definition are being met per client per service, it works with all TV services to help session resource management.

11) Offer management

Maintains definition of every offer and their mapping to content such as linear channels and VOD assets.

12) Authorization validation

Validates entitlement of content per account.

13) Headend purchase

Enables transaction purchase of a one-time offer of VOD or linear content and maintains transactional history of all one-time purchases of VOD or linear content.

14) Access criteria manager

For the DVB live programme channel, the access standard used by the CAS system is defined by this microservice. The access standard is sent to the terminal device by the broadcasting entitlement control message (ECM) data of the CAS. After receiving the ECM, the terminal first determines whether the access criteria (AC) condition is satisfied. If it is satisfied, the ECM will be sent to the smartcard to obtain the descrambling CW. On the other hand, if the AC condition is not satisfied, the CW cannot be obtained, and the programme cannot be descrambled.

8.3 Advertisement and metadata capability

The advertisement and metadata capability provides operational support services for various services of the application layer, including advertising decision system, advertising proxy, advertising reporter, advertising library, advertising manifest management, cloud metadata aggregation, and cloud UX object creation.

1) Ad decision system

Microservices, based on service provider input as well as properties of both advertisement primary content and user, decide which advertisements to send to which users per content.

2) Ad proxy

Proxy of the Ad that receives requests for targeted Ads from Ad stitcher and sends them to Ads for targeted Ad selection.

3) Ad reporter

Microservice that collects data on inclusion and viewing of advertisement attached to a particular content from a client for supporting Ads microservice and analytics.

4) Ad library

The library of transcoded advertisements contents ready for inclusion as part of content viewing.

5) Ad manifest management

Microservice that coordinates the whole content and advertisement manifest list creation process.

6) Cloud metadata aggregation

Metadata includes all programme metadata such as linear, VOD and DVR. Based on the request of a specific UI page, it calls all relevant microservices and aggregates responses to enable cloud UX object creation to create relevant UX objects.

7) Cloud UX object creation

UX object refers to all elements to be displayed on UI interface, such as programme list, buttons, pictures, etc. This microservice understands UI state machine and based on navigation by users, creates all UX objects required in the presentation of the next page to the user.

8.4 Content distribution capability

Content distribution capability is responsible for providing content distribution services for media-based services, including microservices such as content distribution network (CDN) routers, edge servers, central caches, and origin servers.

1) CDN router

Redirects each device to closed edge server assuming vacant capacity to implement load balancing.

2) Edge server

Supports download of IP content including manifest files and chunks.

3) Central cache server

The secondary cache to support edge servers when content is no longer in the edge cache.

4) Origin server

The initial cache of newly ingested contents, all central cache and edge servers will get contents from the origin server and cache contents locally.

8.5 Security capability

Security capability provides security related capabilities for broadcast and converged media services, including microservices such as multi digital rights management (DRM) registration, multiDRM server, key management, licence generation services, authorization object generator, content watermark manager, content watermark ingesting and content watermark detector.

1) MultiDRM registration

Collects all relevant secure properties of the devices. Adds device to account and registers a unique DRM device ID with the identity management service and activates device with data needed to activate video services securely.

2) MultiDRM server

Microservice responsible for validation of authorization object and preparation of parameters for licence generation for each DRM family. The microservice receives content encryption keys from key management microservice and interacts with licence generator to generate licences. For licence parameters generation, multiDRM server may receive data from other sources.

3) Key management

Microservice responsible for generating and storing randomly generated keys that are used for content encryption and licence generation services. It provides content encryption keys to content just-in-time encryption microservice and to multiDRM server for access by licence generator.

4) Licence generator

Microservice responsible for licence generation for a specific DRM type or other DRM types (e.g., PlayReady, FairPlay, Widevine).

5) Business rule generator

Microservice responsible for generation of authorization object/business rules. This object contains parameters necessary for licence generation for specific device/client for particular content by a particular DRM vendor.

6) Content watermark manager

Microservice responsible for managing watermarking contents for further traceability.

7) Content watermark ingesting

Microservice responsible for secure incorporation of content watermark ingestion in the content. Usually, the embedded watermark information is invisible and should not affect the user's normal viewing experience.

8) Content watermark detector

Microservice that detects and analyses marking of a suspect content. The watermark information stored in the watermark management system is compared against the detected watermark.

9) Just-in-time encryptor

Encrypts OTT packets of all supported formats such as HTTP dynamic streaming (HDS), HTTP smooth streaming (HSS), HTTP live streaming (HLS), dynamic adaptive streaming over HTTP (DASH), and extensible formats.

10) CAS server

Manages CAS related products, keys, and authorizations.

11) Entitlement management message generator (EMMG)

The entitlement management message (EMM) is generated and sent to the CAS component of the client and stored on the client. The EMM mainly includes authorization data and other control information.

12) Entitlement control message generator (ECMG)

The ECM is generated and sent to the CAS component of the client. The ECM carries the access criteria (AC) data and the video scrambling CW. If the access criteria in the ECM is verified, and the terminal has the relevant video access authorization, then the CW can be decrypted and output for content descrambling.

8.6 Data analysis capability

Data analysis capability provides big data support for broadcast and converged media services, including microservice as buffering, data cleansing and normalization, whitelisting, object storage, data enrichment, structuring and data access, business analytics and reporting, etc.

1) Buffering

Components such as Kafka or Kinesis are streams that buffer data in an auto-scaling fashion to ensure that logs or events are not lost even during peaks.

2) Data cleansing and normalization

Ensures all incoming field names are in an appropriate and consistent format.

3) Whitelisting

The microservice sets filtering conditions and accepts only names and values of fields required for some AI applications.

4) Object storage

The microservice implements object storage that supports massive cheap and reliable storage of raw data and structured data.

5) Data enrichment

Microservices that enrich the data by either calling external APIs (e.g., IP to location) or creating new data fields by aggregating and/or combining several data fields.

6) Structuring and data access

Process of transforming of raw data to some compressed columnar format and enabling access to the data via fast structured query language (SQL) queries such as usage of Hive or other similar tools.

7) Business analytics and reporting

Descriptive applications that read structured data using SQL and provide descriptive visual or textual information using tools such as Tableau.

8) Log analytics

Capability to store logs in a tool such as Elasticsearch for fast searching of log lines using a UI tool. These tools are generally used for system diagnostic purposes.

9) Database (DB) exporter

The microservice exports database fields to cloud storage of data lake for use in analytics.

10) Message queue and forwarding

Component such as Kafka receives real-time events and forwards them to the streaming ingest component of data lake.

8.7 Data driven application and artificial intelligence (AI) capability

The data driven application and AI capability provides support for the intelligentization of the broadcast and converged media service cloud, including streaming AI applications and Batch AI applications.

1) Streaming AI applications

Real-time AI applications based on real-time algorithms using tools such as Spark streaming, Flink or Storm, which are used mainly for operational management, such as video quality monitoring, data log analysis, and fault alarming.

2) Batch AI applications

Non-real time Batch application based on non-real time AI algorithms using tools such as Spark. Mainly used for personalized applications, service security, business analysis and so on. For example, data-driven advertising, programme recommendation, UI personalization, anti-piracy, business analysis, video content analysis, user behaviour analysis, and so on.

8.8 Billing and payment capability

The billing and payment capability provides support for payment processing, time tracking, automatic bill delivery, etc.

1) Payment processing

Microservices that accept partial, future, recurring, automatic, one-time, and overpayments via credit card, online or mobile.

2) Billing and payment management

Microservices that manage the bills, contact details, banking information, payment histories, time tracking, and other data.

9 Abstraction layer

The abstraction layer provides multiple abstraction functions by deploying the API gateway, which is the entry point for upper applications. In microservices architecture, applications and services are composed of smaller, exchangeable components, and these components need a way to find and communicate with one another. This is where API gateways come in. Instead of calling microservices directly, applications call the API gateway, which forwards the call to the appropriate microservices in the ability layer.

In CBCMS abstraction layer, The API gateway should also perform other cross-cutting functions such as traffic management, authorization and access control, throttling, monitoring, and API version management. These modules implement abstraction and encapsulation of different capabilities and provide the upper-layer software with interfaces used to invoke the corresponding capabilities.

9.1 Abstraction model

The abstraction layer is required to deploy an API gateway that can prevent potential problems with exposing microservices directly to the application layer. The API gateway can handle all the tasks involved in accepting and processing up to hundreds of thousands of concurrent API calls.

API gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. APIs act as the "front door" for applications to access data, business logic, or functionality from backend services.

Figure 3 shows the principle of the abstraction model.

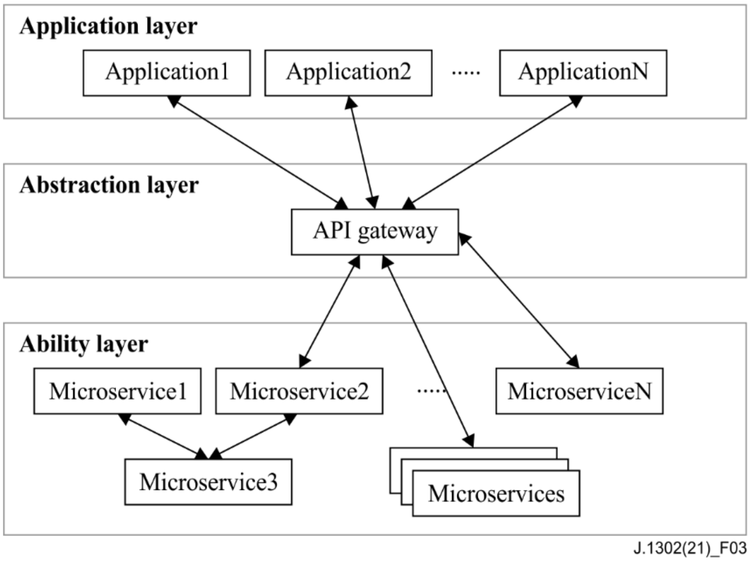

Figure 3 – Principle of the abstraction model

API gateway aggregates the application requests to and from the dependency microservice instances. The client can make one response, which the gateway can break into multiple service requests, saving the bandwidth of having the client make all the calls itself. The gateway will also keep track of the requests.

9.2 Traffic management

The traffic management should implement the following functions:

- Route client requests to the services. This provides a single endpoint for clients and helps to decouple clients from services.
- Aggregate multiple requests into a single request, to reduce chattiness between the client and the backend.
- Offload functionality from the backend services, such as secure sockets layer (SSL) termination, authentication, IP restrictions, or client rate limiting.

9.3 Authentication and authorization

From the security point of view, the abstraction layer should handle the authentication and authorization from the external callers to the microservice level. It provides protection, limitation, and control of access to microservices.

- Authorize and verify application requests to microservices. Access to the microservices will be successfully provided given that the application has the right to use the microservices.
- Configure resource-based policies to allow or deny access to microservices.
- Offer flexible and robust access controls.

9.4 Throttling

Throttling is responsible for preventing services from being overwhelmed by too many requests. API gateway throttles requests to API using the token bucket algorithm, where a token counts for a request.

- Provides the configuration of the limit on a steady-state rate and a burst of request submissions.
- Fails the limit-exceeding requests and returns error to client when request submissions exceed the steady-state request rate and burst limits.

9.5 Monitoring

Monitoring is an important part of maintaining the reliability, availability, and performance of API gateway.

- Monitors specified metrics over a period and performs one or more actions based on the value of the metric relative to a given threshold over several periods.
- Creates log groups and log streams, and reports to the log streams any caller's requests and responses.

10 Application layer

The application layer consists of a user interface, service encapsulation and aggregation, playback applications, information applications and management applications. The applications support secure auto-scaling and each service scales independently based on the microservice architecture. Every application comes with its own set of ability layer microservices performing subsets of functionality.

The application layer shall ensure convergence of IP and DVB-C platforms, allowing users to consume contents both by terminal devices connected to DVB distribution network – cable, – and by terminal devices receiving these contents via IP network – managed or unmanaged.

10.1 User interface

The user interface receives requests from the user clients, collects user information and submits it to the service encapsulation and aggregation module. The data includes user terminal information, user network status information, etc., enabling the module to invoke the appropriate application type and provide the appropriate quality of service to the clients.

10.2 Service encapsulation and aggregation

The service encapsulation and aggregation module, according to the request and user information, call the corresponding application, encapsulate the app, the corresponding business presented to the user.

The service encapsulation and aggregation module call the appropriate application according to the request and sends the user terminal information and network status information to the application, which enables the application to call the corresponding microservices.

10.3 Playback applications

10.3.1 Linear and time-shifting service

This service enables the functions including:

- Ingest of a channel lineup and linear schedule by the service provider describing the linear channels and upcoming linear events.
- Discovery of both linear and time-shifted content from a user interface/electronic programme guide (UI/EPG).
- Advanced search and recommendations of linear events.
- Validation and enforcement of the account's capability to view linear content from their device type and location based on authorization and service provider policies.
- Encoding of real-time linear streams for IP consumption.

- Capturing and recording of those streams in real-time to support restart and catchup.
- Just-in-time packaging of the streams in support of separate device types for consumption using multiple stream formats.
- Trick modes, skip forwards and backwards of the linear content.

10.3.2 Cloud DVR service

This service enables:

- Recording of linear content by the client devices.
- Recording of linear streams in the cloud in real-time.
- Storage of the cloud DVR content as a private copy, a unique copy, or a common copy.
- Discovery of all recorded content from a UI/EPG.
- Advanced search of recorded content.
- Upon playback request, validation and enforcement of account's capability to view recorded content from their device type and location based on authorization and service provider policies.
- Just-in-time packaging of the recorded streams in support of separate device types for consumption using multiple stream formats.
- Trick modes, skip forwards and backwards of the recorded content.

10.3.3 VOD service

This service enables all functions required for a VOD service including:

- Ingest and curation of a VOD catalogue by the service provider describing the VOD assets.
- Discovery of VOD assets from a UI/EPG.
- Advanced search and recommendations of VOD assets.
- Validation and enforcement of the account's capability to view VOD assets from their device type and location based on authorization and service provider policies.
- Encoding of VOD streams for IP/OTT consumption.
- Just-in-time packaging of the streams in support of separate device types for consumption using multiple stream formats.
- Trick modes, skip forwards and backwards of the VOD content.

10.3.4 Advertisement service

This service enables all functions required for advertisement including:

- Selecting the appropriate advertisements per individual to attach to requested VOD, cloud DVR or linear content.
- Creating a new manifest file that includes both the primary content and the targeted Ads based on some Ad inclusion strategy including pre-roll, mid-roll, or post-roll.
- Reporting Ads consumption by users.

10.4 Information applications

10.4.1 UI/UX API service

This service enables all functions required for UI/UX API service including:

- Definition of UI/UX for client.
- Aggregation of UI/UX call across relevant microservices.

10.4.2 Analytics service

This service enables all functions required for analytics including:

- Collecting logs and events from all microservices.
- Receiving exports of all databases.
- Normalizing and enriching data.
- Structuring data to a compressed format and enabling secure access of data by AI and non-AI applications.

10.4.3 Data application services

This service includes training and testing all AI data applications including:

- Business analytic applications.
- Anti-fraud security applications.
- Operational analytics applications.
- Personalization applications.

10.5 Management applications

10.5.1 Identity management service

This service enables all functions required for identity management including:

- Management per account of devices.
- Management per account of users.
- Management per account of offers.
- Management per account of individual policies.

10.5.2 Policy management service

This service enables all functions required for policy management including:

- Describing all policies of a service provider.
- Communicating policies to relevant microservices for enforcement of those policies.
- Examples of policies:
  - Concurrency restrictions per account, devices, contents.
  - Geographical restrictions per account, devices, contents.

10.5.3 Offer management service

This service enables all functions required for offer management including:

- Authorization of accounts to various offers – both subscription and transactional offers.
- Definition and enforcement of entitlements/offers per contents such as linear channels or VOD offers.
- Enforcement of parental rating per user per content.

10.5.4 Billing and payment management

This service enables all functions required for billing and payment management including:

- Billing applications.
- Payment applications.
- Refund applications.

10.5.5 Secure session management

This service enables all functions required for registering new devices to the system and performing session management including:

- Initial authentication of user via identity provider.
- Registering new user to system.
- Ongoing authentication of user before accessing services.
- Providing other service access to core account information such as device ID and account ID.

10.5.6 Security service

This service enables all functions required for security including:

- Encryption of linear, VOD and cloud DVR content.
- Insertion of watermarking within linear, VOD and cloud DVR content.
- Authentication of all devices before enabling them access to any OTT video service.
- Support of both DRM and conditional access (CA).
- Support of secure DRM licences upon authorization of an account and device.

11 Management tools

The management tools provide a standard platform for operators to meet the principles of fast velocity, automatic testing, and deployments. This layer is composed of a series of components, which is required to register and orchestrate microservices, manage basic functions, and other components' managements for the whole architecture.

11.1 Microservice orchestration

The microservice orchestration provides service choreography and orchestration:

- Implementing the orchestration of calls to multiple microservices, so that the consumer obtains all the functionality that they need without having to make many calls that do not give them any value.
- Offering functionality, facilitating the control and visualisation of the interactions between the microservices.
- Supporting synchronous performance if necessary.

11.2 Service discovery

In a microservices application, the set of running service instances changes dynamically. Instances have dynamically assigned network locations. Consequently, for a client to request a service, the client must use a service discovery mechanism.

Service registry is the key part of service discovery. The service registry is a database of available service instances. The service registry provides a management API and a query API. Service instances are registered with and deregistered from the service registry using the management API. The query API is used by system components to discover available service instances.

Service registry provides a database containing the network locations of service instances. Services register themselves to the registry during start up, and when service instances change, the information needs to be updated. It needs to be highly available as it consists of a cluster of servers that use a replication protocol to maintain consistency.

Commonly, the service registry module provides a REST API for registering and querying service instances. A service instance registers its network location using a POST request. Every certain time it must refresh its registration using a PUT request. A registration is removed by either using an HTTP DELETE request or by the instance registration timing out. Client can retrieve the registered service instances by using an HTTP GET request.

The CBCMS service discovery model is shown in figure 4.

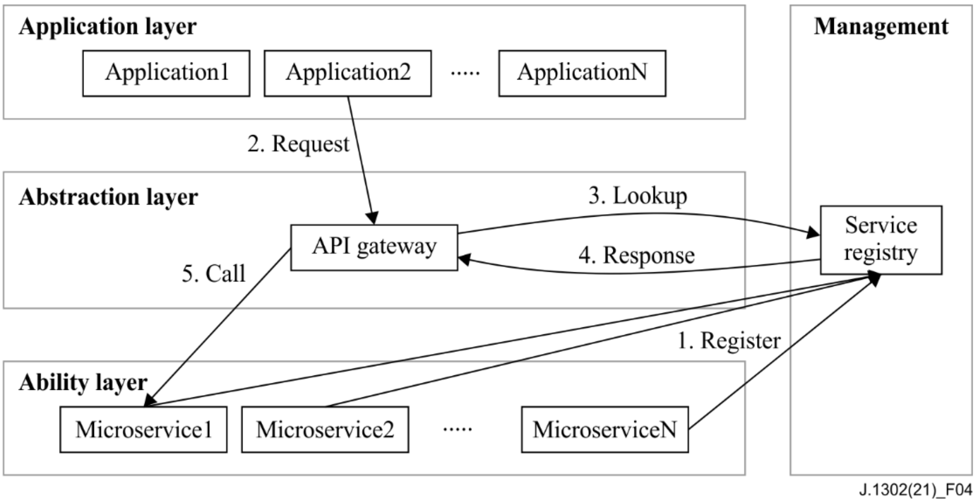

Figure 4 – Service discovery model

1) Each instance of the microservices register to the registry at start up.

2) An application performs a request to the API gateway.

3) The API gateway lookup the relevant microservice instance.

4) The service registry responds with the address of the microservice.

5) The API gateway processes the call and forwards it to the requested microservices.

11.3 Lifecycle management

Lifecycle management is specifically related to auto-scaling and self-healing. Auto-scaling is needed to create new instances when traffic increases and destroy unneeded ones when traffic decreases. Self-healing identifies and recreates failed service instances. Both these features make certain assumptions about the component's internal business logic and need to be configured correctly for each component.

11.4 Problem management

When an incident occurs or when a change is needed, the incident management support is notified and the application is fixed by either restarting the VMs, containers and/or applications, or by adding or removing capacity.

11.5 Dark launch

Dark launch helps developers to test the beta version, ensuring smooth rollout of new features. Once new features become mature and stable, a formal version is released for all users to use.

11.6 Log analytics

Log analytics is the process of analysing aggregated log data to extract knowledge from them, which helps operators collect and analyse log data. It provides the services for editing, querying and analysing log data in CBCMS.

Bibliography

[b-ITU-T J.205] Recommendation ITU-T J.205 (2012), *Requirements for an application control framework using integrated broadcast and broadband digital television*.

[b-ETSI EN 301 192] ETSI EN 301 192 V1.6.1 (2015), *Digital Video Broadcasting (DVB); DVB specification for data broadcasting*.

[b-ETSI TS 102 809] ETSI TS 102 809 V1.3.1 (2017), *Digital Video Broadcasting (DVB); Signalling and carriage of interactive applications and services in Hybrid broadcast/broadband environments*.

[b-GB-SEP] GB/T 28160 (2011), *Specification of electronic programme guide for digital television broadcasting (in Chinese).*

[b-GB-SSI] GB/T 28161 (2011), *Specification of service information for digital television broadcasting (in Chinese).*

[b-ISO/IEC 14496-12] ISO/IEC 14496-12 (2015), *Information technology – Coding of audio-visual objects – Part 12: ISO base media file format*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems, and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	**Cable networks and transmission of television, sound programme and other multimedia signals**
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2021