International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.2025

(07/2012)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

Next Generation Networks – Frameworks and functional architecture models

Functional architecture of the next generation network service integration and delivery environment

Recommendation ITU-T Y.2025

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	**Y.2000–Y.2099**
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Numbering, naming and addressing	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.2025

Functional architecture of the next generation network service integration and delivery environment

Summary

Recommendation ITU-T Y.2025 defines the functional architecture of next generation network service integration and delivery environment (NGN-SIDE), which includes the definition of functional entities, reference points, service procedures and interconnection between different NGN-SIDEs.

This Recommendation builds on the NGN-SIDE framework, general requirements, capabilities, and requirements related to service interfaces described in Recommendation ITU-T Y.2240.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T Y.2025	2012-07-29	13

Keywords

Functional architecture, functional entities, NGN, NGN-SIDE, reference points, resource adaptation, service creation, service delivery management, service execution, service integration.

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, 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 TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Y.2025

Functional architecture of the next generation network service integration and delivery environment

1 Scope

This Recommendation defines the functional architecture of NGN service integration and delivery environment (NGN-SIDE), which includes the definition of functional entities, reference points, service procedures and interconnection between different NGN-SIDEs.

The Recommendation builds on the NGN-SIDE framework, general requirements, capabilities, and requirements related to service interfaces described in [ITU-T Y.2240].

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 Y.2012] Recommendation ITU-T Y.2012 (2010), *Functional requirements and architecture of next generation networks*.

[ITU-T Y.2221] Recommendation ITU-T Y.2221 (2010), *Requirements for support of ubiquitous sensor network (USN) applications and services in the NGN environment*.

[ITU-T Y.2233] Recommendation ITU-T Y.2233 (2010), *Requirements and framework allowing accounting and charging capabilities in NGN*.

[ITU-T Y.2240] Recommendation ITU-T Y.2240 (2011), *Requirements and capabilities for next generation network service integration and delivery environment*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 application [b-ITU-T Y.2261]: A structured set of capabilities, which provide value-added functionality supported by one or more services, which may be supported by an API interface.

3.1.2 application network interface [ITU-T Y.2012]: Interface which provides a channel for interactions and exchanges between applications and NGN elements. The ANI offers capabilities and resources needed for the realization of applications.

3.1.3 application programming interface [b-ITU-T I.312]: An API provides a set of interfaces from an application environment to an execution environment. The execution environment provides services to the application environment.

3.1.4 functional architecture [ITU-T Y.2012]: A set of functional entities and the reference points between them used to describe the structure of an NGN. These functional entities are separated by reference points, and thus, they define the distribution of functions.

3.1.5 functional entity [ITU-T Y.2012]: An entity that comprises an indivisible set of specific functions. Functional entities are logical concepts, while groupings of functional entities are used to describe practical, physical implementations.

3.1.6 next generation network (NGN) [b-ITU-T Y.2001]: A packet-based network able to provide telecommunication services and able to make use of multiple broadband, QoS-enabled transport technologies and in which service-related functions are independent from underlying transport-related technologies. It enables unfettered access for users to networks and to competing service providers and/or services of their choice. It supports generalized mobility which will allow consistent and ubiquitous provision of services to users.

3.1.7 NGN service integration and delivery environment (NGN-SIDE) [ITU-T Y.2240]: An open environment in NGN integrating resources from different domains and delivering integrated services to applications over NGN.

NOTE – These domains include, but are not limited to, telecommunication domain (e.g., fixed and mobile networks), Internet domain, broadcasting domain and content provider domain.

3.1.8 NGN service stratum [b-ITU-T Y.2011]: That part of the NGN which provides the user functions that transfer service-related data and the functions that control and manage service resources and network services to enable user services and applications.

3.1.9 NGN transport stratum [b-ITU-T Y.2011]: That part of the NGN which provides the user functions that transfer data and the functions that control and manage transport resources to carry such data between terminating entities.

3.1.10 reference point [ITU-T Y.2012]: A conceptual point at the conjunction of two non-overlapping functional entities that can be used to identify the type of information passing between these functional entities.

NOTE – A reference point may correspond to one or more physical interfaces between pieces of equipment.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

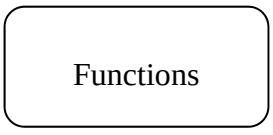
ANI	Application Network Interface
API	Application Programming Interface
BPEL	Business Process Execution Language
CDR	Call Detail Record
CP	Content Provider
CPU	Central Processing Unit
CSN	Converged Service Network
FE	Functional Entity
FTP	File Transfer Protocol

GUI	Graphics User Interface
HTTP	Hyper Text Transfer Protocol
IdM	Identity Management
ISDN	Integrated Services Digital Network
NGN	Next Generation Network
NGN-SIDE	NGN Service Integration and Delivery Environment
NNI	Network-Network Interface
OSE	Open Service Environment
PLMN	Public Land Mobile Network
POJO	Plain Old Java Objects
PSTN	Public Switched Telephone Network
QoS	Quality of Service
REST	Representational State Transfer
SLA	Service Level Agreement
SMS	Short Message Service
SNI	Service Network Interface
SOAP	Simple Object Access Protocol
SP	Service Provider
UNI	User Network Interface
USN	Ubiquitous Sensor Network

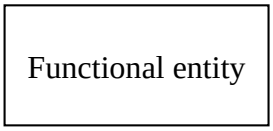
5 Conventions

In this Recommendation:

Functions: In the context of NGN-SIDE architecture, "functions" are defined as a collection of functionalities. It is represented by the following symbol:

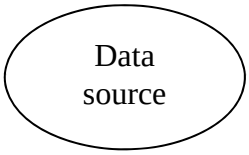


Functional entity: In the context of NGN-SIDE architecture, a "functional entity" is defined as set of one or more functions. It is represented by the following symbol:



Functional group: In the context of NGN-SIDE architecture, a "functional group" is defined as a group of related functional entities. It is represented by the symbol as "functions" shown above.

Data source: In the context of NGN-SIDE architecture, "data source" is defined as particular source of local storage of data (e.g., repository specific to some functional entity). It is represented by the following symbol:



6 Overview of NGN-SIDE

6.1 NGN-SIDE general characteristics

NGN-SIDE [ITU-T Y.2240] provides an environment for service creation, service execution, and service delivery management in order to support the integration of capabilities from different domains over NGN. NGN-SIDE also provides resource brokering and resource adaptors capabilities.

The general characteristics of NGN-SIDE are described as follows:

1) NGN-SIDE provides exposure of resources (including service enabler, content, applications and virtualized resources) from different domains (NGN, fixed networks (e.g., PSTN/ISDN), mobile networks (e.g., PLMN), Internet, content providers, broadcast networks (e.g., cable networks, etc.) to applications in a secure and controlled way.
2) NGN-SIDE provides a service creation environment including capabilities for application development, resource repository, and testing environment.
3) NGN-SIDE provides a configurable, scalable, reliable and virtualized service execution environment including capabilities for resource registration, access control, static and dynamic service orchestration capability, resource manager, and service dispatching.
4) NGN-SIDE provides a manageable and configurable service delivery management environment including capabilities for the management of role related information, charging, context management, content management, policy management, application provisioning.
5) NGN-SIDE provides capabilities for resource brokering and resource adaptor including adaptors for NGN and adaptors for non-NGN.
6) NGN-SIDE provides mechanisms to support USN applications [ITU-T Y.2221].

6.2 Position of NGN-SIDE in NGN architecture

Figure 6-1 shows position of service integration and delivery environment in NGN functional architecture [ITU-T Y.2012].

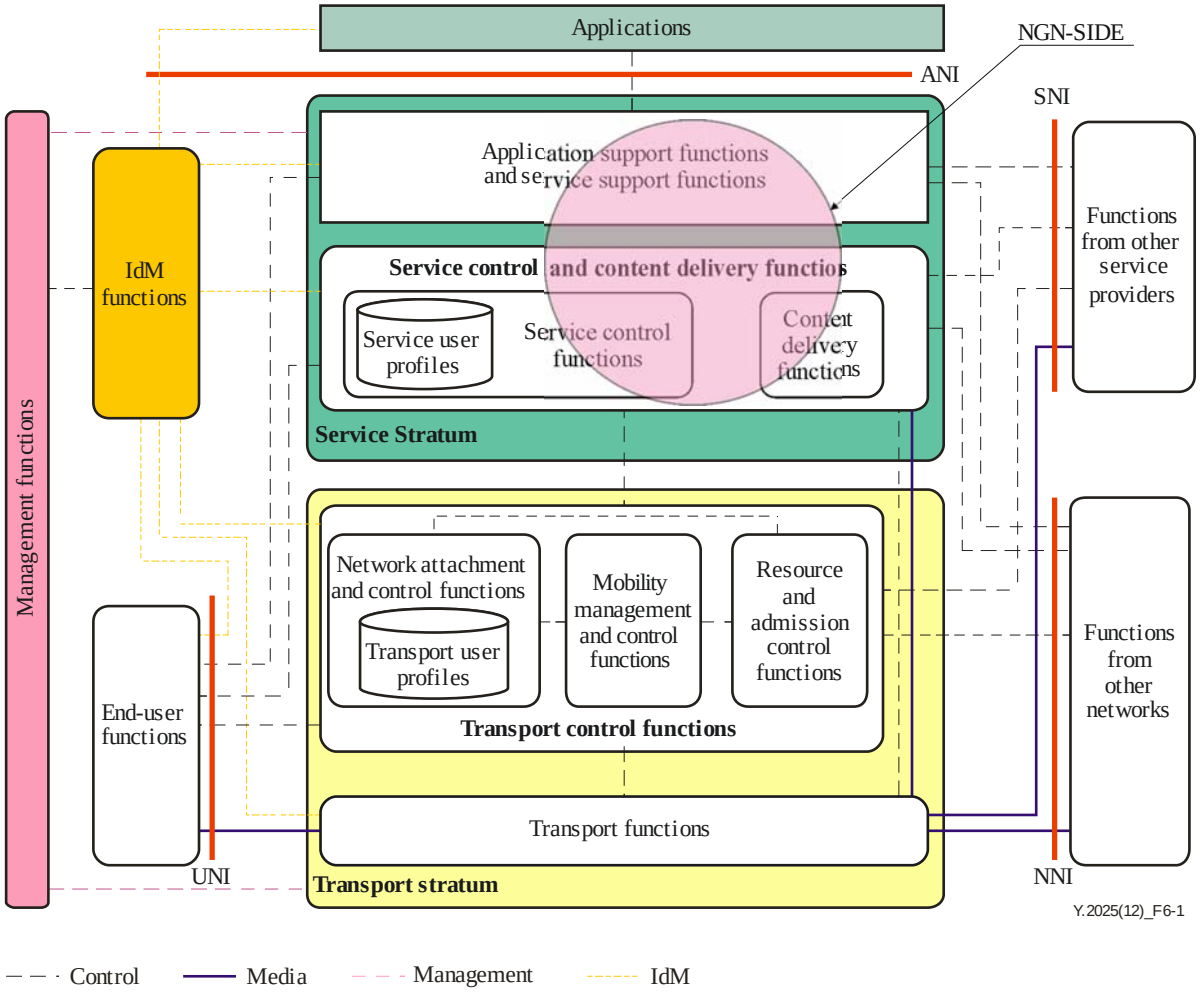


Figure 6-1 – The position of NGN-SIDE in NGN architecture

As shown in Figure 6-1, NGN-SIDE functions reside in the NGN service stratum [ITU-T Y.2012].

6.3 Introduction to NGN-SIDE functional architecture

Figure 6-2 shows the general architecture of NGN-SIDE. The detailed functional architecture is specified in clause 7.

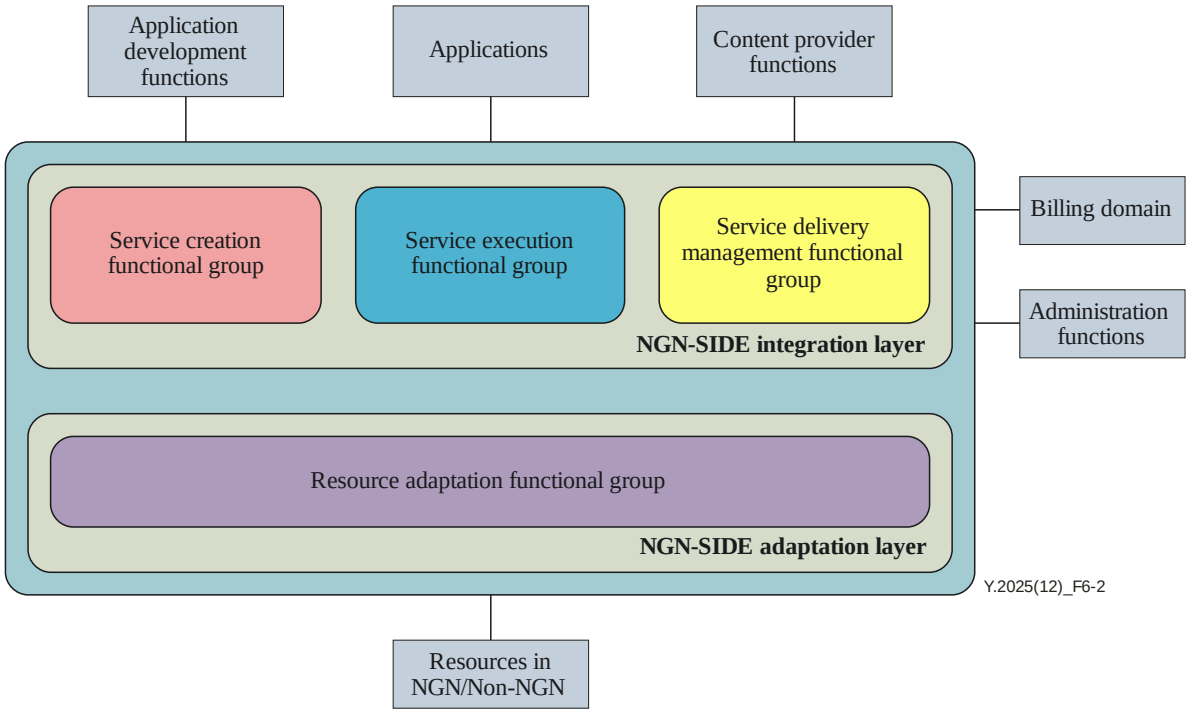


Figure 6-2 – NGN-SIDE functional architecture overview

As shown in Figure 6-2, applications, application development functions, content provider functions, billing domain, administration functions and resources in NGN/non-NGN interact with NGN-SIDE.

6.3.1 NGN-SIDE functions

In line with [ITU-T Y.2240], NGN-SIDE architecture is composed of two layers:

– NGN-SIDE integration layer

– NGN-SIDE adaptation layer.

NGN-SIDE integration layer supports the following functional groups:

– service creation functional group

– service execution functional group

– service delivery and management functional group.

The service creation functional group provides entities to realize an application development environment for application developers. Application developer interacts with service creation functional group to create applications.

The service execution functional group provides entities to support the service execution environment. Service execution receives requests from applications and executes service requests.

The service delivery management functional group provides management support for ensuring proper functioning of the service creation and service execution functional groups and providing associated delivery functionalities.

The NGN-SIDE adaptation layer supports resource adaptation functional group which provides entities to realize resource brokering and resource adaptors. The resource brokering function identifies the appropriate resources from the NGN-SIDE resource based on NGN-SIDE users' request and mediates among resources and applications. The resource adaptor function performs

adaptation of resources from the NGN-SIDE resource provider. The adaptation functional group interacts with different resources offered by NGN and non-NGN.

6.3.2 Functions external to NGN-SIDE

An application developer accesses the FEs in the service creation functional group via application development functions in order to create and test applications.

When an (in-house or third party) application intends to consume the resources integrated in NGN-SIDE, the application sends a resource request to the service execution functional group. The service execution functional group interacts with the related FEs in the service execution functional group and service delivery management functional group to retrieve information related to the processing of request accessing resource in NGN and/or non-NGN.

The resource request is sent to the adaptation functional group to invoke the requested resource. The adaptation functional group brokers and adapts the appropriate resource.

After the resource execution is finished, the response is sent back to the application via the service execution functional group.

The content provider functions enable content provider to make available different types of content to NGN-SIDE.

The billing domain receives and processes charging information from the NGN-SIDE charging functions. As a business support function, the billing domain is outside the NGN-SIDE functional architecture.

The administration functions enable administrator to perform several functions including content management and application life cycle management.

7 Functional architecture of NGN-SIDE

This clause specifies the NGN-SIDE functional architecture with the description of the NGN-SIDE functional entities (FEs) being provided in the following sub-clauses. The description of the NGN-SIDE reference points is provided in clause 8.

The functional architecture of NGN-SIDE is shown in Figure 7-1.

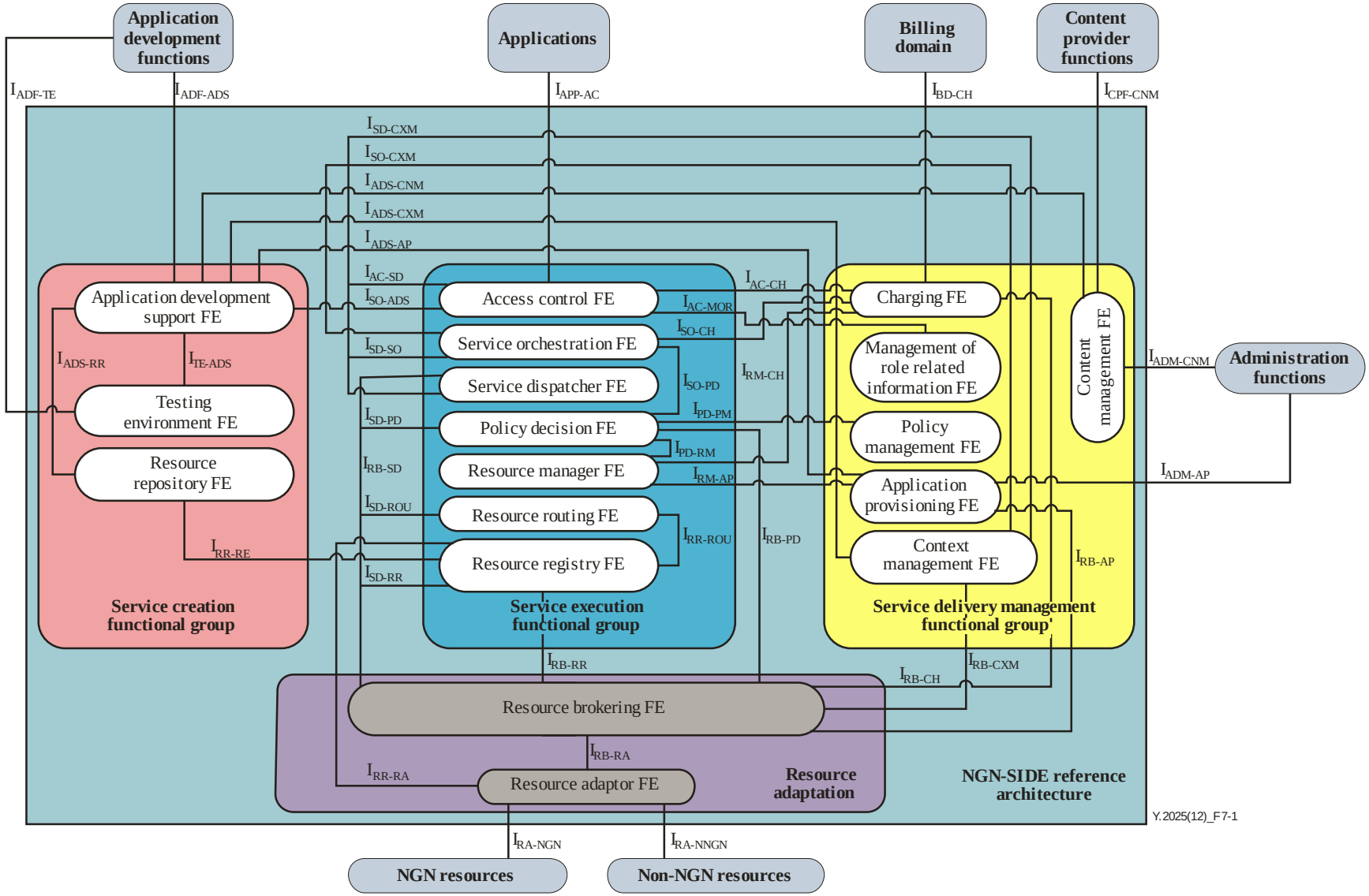


Figure 7-1 – NGN-SIDE detailed functional architecture

The service creation functional group enables application developers to create and test applications. The service creation functional group includes the following functional entities:

– application development support FE
– testing environment FE
– resource repository FE.

Service execution functional group enables applications to access exposed resources. Service execution functional group includes the following functional entities:

– access control FE
– policy decision FE
– resource manager FE
– service dispatcher FE
– service orchestration FE
– resource routing FE
– resource registry FE.

The service delivery management functional group supports service execution and service creation mechanisms. The service delivery management functional group interacts with different external functions such as content provider functions, administration functions and billing domain for content handling, administrative and billing, respectively. The service delivery management functional group includes the following functional entities:

– content management FE
– management of role related information FE
– policy management FE
– application provisioning FE
– charging FE
– context management FE.

The adaptation functional group performs resource mediation and adaptors function to access resource in different domains as well as enables different resources to interact with applications. Adaptation functional group includes the following functional entities:

– resource brokering FE
– resource adaptor FE

7.1 NGN-SIDE service creation functional group related functional entities

7.1.1 Application development support FE

The application development support FE enables application developers to develop applications using resources exposed by NGN-SIDE.

Figure 7-2 shows a functional view of application development support FE.

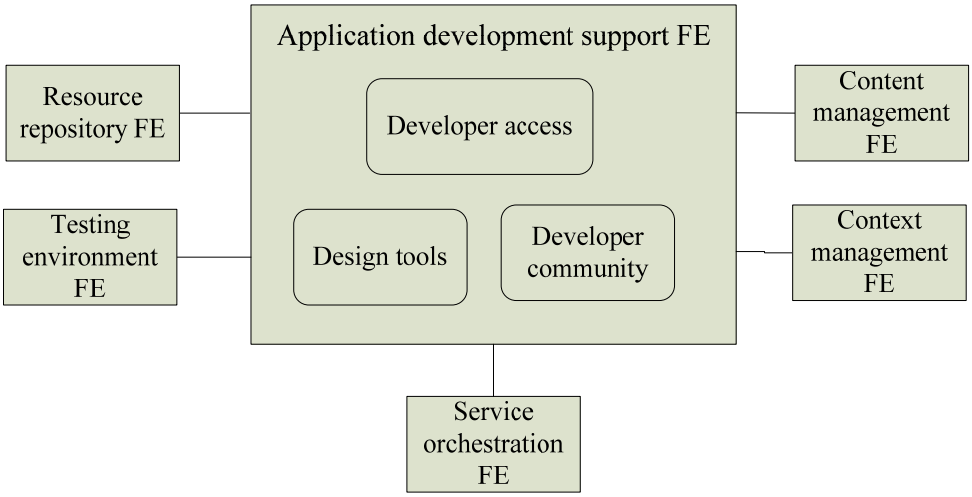


Figure 7-2 – Application development support FE functional view

Application developers can access this FE to develop applications using online and/or offline design tools. The design tools facilitate application development and provide functions including drag and drop, automatic code generation, etc. When developing an application, the application development support FE interacts with the resource repository FE to achieve the development interface information of the resources used in the application, and also interacts with the content management FE to achieve the information of the content used in the application.

For the applications requiring context awareness information, the context management FE exposes the related context information which is used in the application creation procedure.

Developers also can develop composite resources with the application development support FE. Composite resources are stored in the service orchestration FE to be accessible as a resource for other applications.

The developer community function in this FE is an optional functionality which supports a developer forum to discuss and exchange their ideas, and support collection and exposure of customer feedback and ratings about applications, ratings about resources and tools, and provide self-service functions. Furthermore, the developer community provides an entrance for developers to NGN-SIDE. It supports functions of developer registration. In addition, developers can use it to download development documents.

NOTE – The capability of the developer community function can be offered by an independent web portal service. This Recommendation assumes this function is integrated into NGN-SIDE.

7.1.2 Resource repository FE

The resource repository FE provides functionalities for the storage of resource interface description information related to the resources. Application developers use the resource interface to create applications.

Figure 7-3 shows a functional view of resource repository FE.

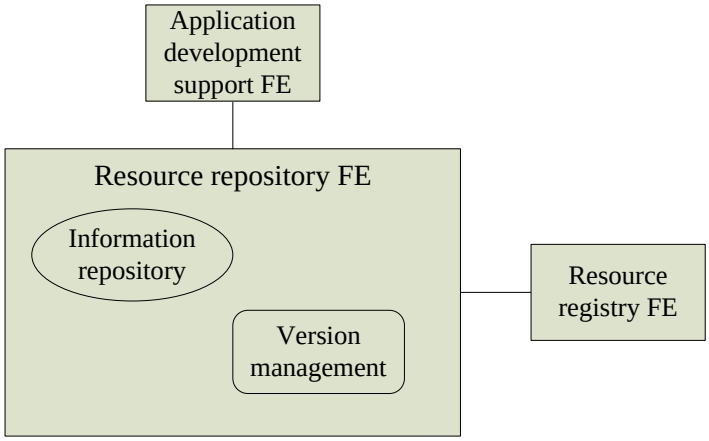


Figure 7-3 – Resource repository FE functional view

The resource repository FE interacts with the resource registry FE for getting resource interface description information.

When a resource is registered to the resource registry, the resource information related to application development such as resource interface description information is stored in the resource repository. The resource interface is exposed to application developers.

The resource repository FE also provides resource version management mechanism to keep synchronization of resource information registered at resource registry FE.

7.1.3 Testing environment FE

Figure 7-4 shows a functional view of testing environment FE.

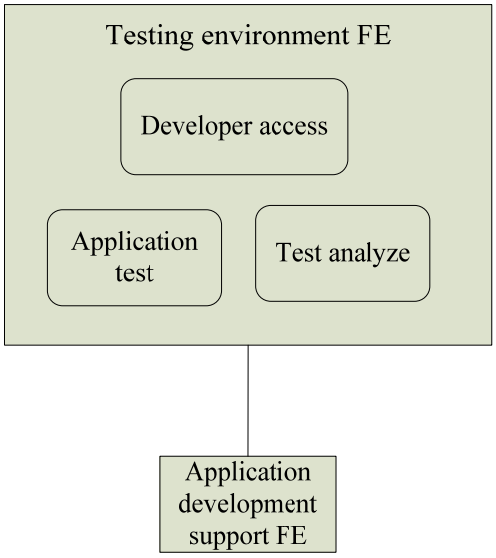


Figure 7-4 – Testing environment FE functional view

Application developers can access this FE to test the created application to ensure its normal running before the application is published. The testing operations include tracing and debugging.

The developer access function can check the developer's access permission and provide a GUI including an overview of the developer's own application such as latest test status.

After the developer's access, the application test function provides:

1) Multiple simulation roles including network, applications, end users, etc.
2) Online/offline testing tools for tracing and debugging
3) Real-world network uncertainties simulation, e.g., variable response time delays
4) Configuration of individual subscriber and large groups of subscribers for simulation
5) Traffic monitoring capability between application and simulator

The test analysis function can provide a testing report for the developers to get detailed information, and may highlight some part with suggestions which generated automatically.

The testing environment FE interacts with the application development support FE to provide simulation of resources exposed by NGN-SIDE for application developers to test their applications.

NOTE – Application testing activities should not impact the regular operations of NGN-SIDE.

7.2 NGN-SIDE service execution functional group-related functional entities

7.2.1 Resource registry FE

The resource registry FE provides resource registration, resource discovery and resource governance functionalities. Additionally, it maintains a catalogue of the registered resources.

Figure 7-5 shows a functional view of resource registration FE.

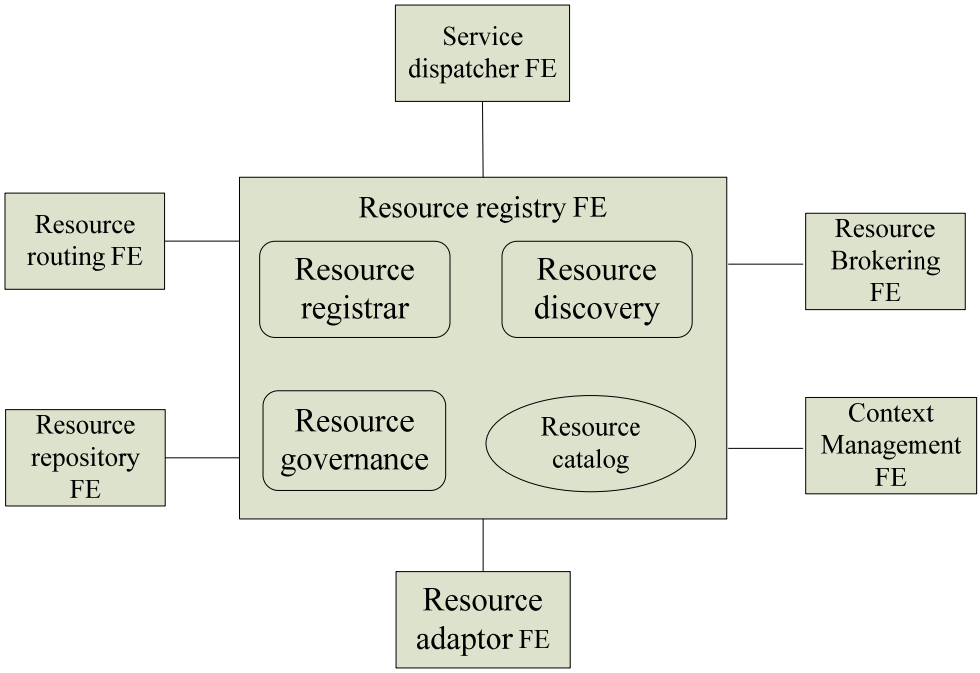


Figure 7-5 – Resource registry FE functional view

The resource registrar function processes the resource registration request from resource providers. The resource information is stored in resource catalogue. When the resource needs unregistration, the resource registrar removes the subscription of the resource from the local catalogue and notifies the resource repository FE, resource brokering FE and resource adaptor FE accordingly.

During the resource registration process, the resource registry interacts with the resource adaptor FE and the resource repository functional entity.

When the resource is registered at the resource registry FE, the resource registry FE:

– sends an adaptation request including resource registration information towards the resource adaptor FE to set up a corresponding adaptation function for this resource;

– sends resource interface description information to the resource repository FE to store an interface description for application development purposes;

– sends resource information to the resource brokering FE and the resource brokering FE schedules resources accordingly;

– provides resource routing-related information to the resource routing FE.

When the service dispatcher FE sends a request to query resource registration information, the resource discovery function consults the resource catalogue to locate the requested resource according to the resource information received and sends the outcome accordingly.

The resource governance function manages resource life cycle and synchronizes information with context management FE.

7.2.2 Access control FE

The access control capability provides translation of APIs/protocols across different NGN-SIDE service interfaces as well as access from applications to functionalities and/or resources exposed by NGN-SIDE [ITU-T Y.2240].

The access control FE includes the following sub-functions:

– API/Protocol translation

– Authentication and authorization

– Relaying request/response

– Triggering charging event.

Figure 7-6 shows a functional view of the access control FE.

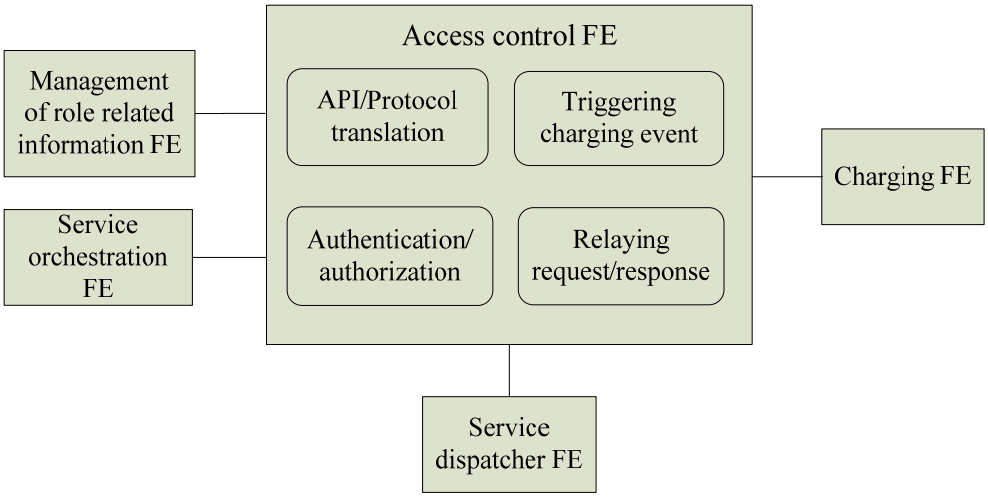


Figure 7-6 – Access control FE functional view

The API/protocol translation function translates the message request from the relevant service API and subsequently the authentication/authorization function authenticates and/or authorizes the message request with the interaction of management of role-related information FE. After authentication/authorization, the triggering charging event sends the charging-related information to the charging FE and finally, the relaying request/response function forwards the request to the service dispatcher FE or towards applications.

API/Protocol translation

This function parses the requests as received from different applications to perform different actions including request authentication, authorization and charging-related functions.

Authentication and authorization

This function authenticates and authorizes the requests from applications with the interaction of management of the role-related information FE.

This sub-function also interacts with the IdM function in NGN to achieve end user, application and service-related identity information for authentication and authorization.

The request is authenticated to identify the application identity securely and further authorized (e.g., the request needs to consume protected resources controlled by the resource provider which has the right of granting application X access to the resource) according to the approval of the resource provider and service level agreements (SLAs).

Triggering charging event

This function interacts with charging FE to perform different charging functions.

In case of online charging, this function will support mechanisms to trigger charging events. In case of offline charging, this function will support mechanisms to generate and deliver CDRs to the charging FE.

Relaying request/response

This function is responsible for forwarding:

– the request/response from application towards target resource

– the request/response from resource towards the target application.

7.2.3 Policy decision FE

The policy decision FE performs all policy rule evaluation related functions. The policy decision FE includes two sub-functions: decision making and synchronization. Additionally, the policy decision FE provides storage for the policies.

Figure 7-7 shows the policy decision FE functional view.

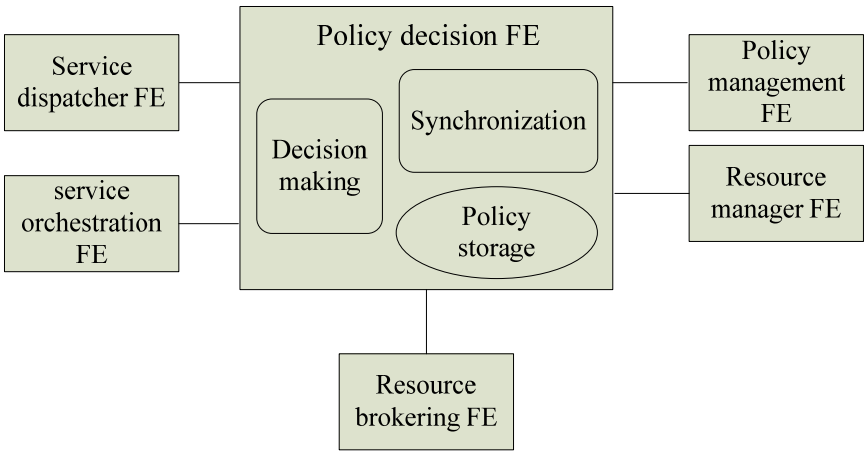


Figure 7-7 – Policy decision FE functional view

Policies include SLAs between NGN-SIDE users and NGN-SIDE resources, end user's preferences and in-house application's resource requirements (related to computing, connectivity and storage resources).

The policy decision FE synchronizes the policies from the policy management FE, and stores the policies in policy storage.

The service dispatcher FE, the service orchestration FE and the resource brokering FE interact with the decision making function of the policy decision FE to acquire the policy decision making results. The service dispatcher FE, the service orchestration FE and the resource brokering FE enforce the policies according to policy decision result. The resource manager FE also interacts with the policy decision FE to manage resources for in-house applications.

7.2.4 Resource manager FE

The resource manager performs the controlling functions for all resources in order to satisfy the application's requirements. During the deployment of NGN-SIDE in-house applications, the application provisioning capability communicates with the resource manager which assigns the resources (including computing, connectivity, and storage) internal to NGN-SIDE according to the application's requirements (SLAs) and the NGN-SIDE provider's policies [ITU-T Y.2240].

Figure 7-8 shows a resource manager FE functional view.

The resource manager includes the following sub-functions:

– Resource creation
– Resource allocation
– Resource release
– Resource tracking and statistical analysis.

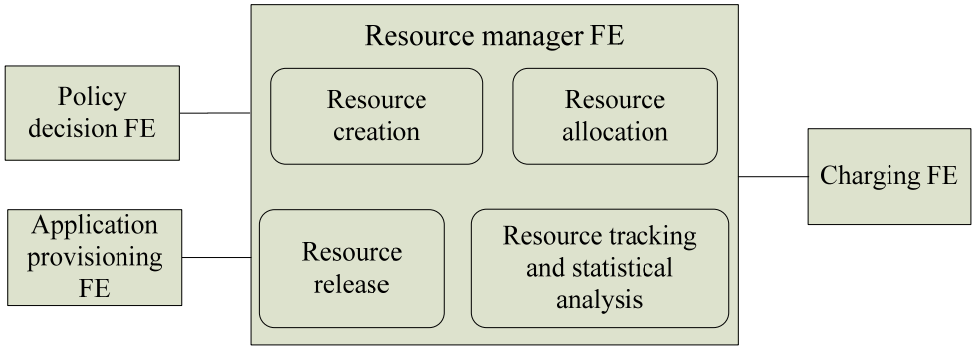


Figure 7-8 – Resource manager FE functional view

Resource creation

Resource creation enables the creation of a resource pool. This function is responsible for resource profiling, including CPU profile, memory profile, network profile and storage profile, according to a resource template, so that different resources can be allocated to different applications in a consistent and systematic way.

Resource allocation

Resource allocation enables the provisioning of resources to applications, including CPU, memory, network and storage resources, among multiple applications on an application server according to defined policies.

Resource release

When an application is uninstalled, the related resources are released. As a result, these resources are added to the resource pool.

Resource tracking and statistical analysis

Resource tracking keeps runtime information about resources in the resource pool, such as available resources at any point of time. Resource information is updated when a particular resource is allocated or released.

Resource statistical analysis is an optional capability which shows the information of a resource's usage at regular intervals. It supports multidimensional analysis, such as all resources' usage by one application, all resources' usage by one application provider, CPU's usage by all applications, memory's usage by all applications, etc.

NOTE – This Recommendation assumes that the resource statistical analysis, which can be an independent capability outside NGN-SIDE, is integrated into NGN-SIDE.

7.2.5 Service dispatcher FE

This capability provides unified message routing and message exchange mechanisms within NGN-SIDE [ITU-T Y.2240]. Message routing and message exchange mechanisms deal with varying qualities of service, depending on application needs, resource performance and resource policies.

Figure 7-9 shows the service dispatcher FE functional view.

Service dispatcher includes the following sub-functions:

– Message routing
– Message transformation
– Message exception handling
– Service logic parsing and processing.

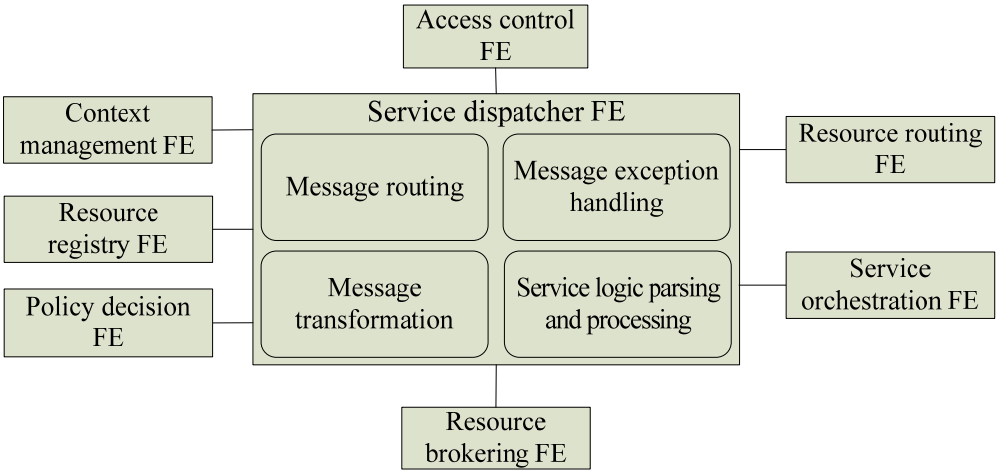


Figure 7-9 – Service dispatcher FE functional view

Message routing

A request message from an application is passed through the access control FE. The message routing function conducts route calculation for requests and responses between NGN-SIDE users and NGN-SIDE resource providers. For selecting a proper resource, the service dispatcher FE accesses information in policy decision FE, resource registry FE, resource routing FE and context management FE. With the resource information, context information and the policy information, the message routing function supports various strategies such as static/deterministic routing, dynamic

routing, content-based routing, rule-based routing, context-based routing, policy-based routing and so forth.

When the destination FE is determined, a new message (using common message structure) is generated and sent to the target functional element. The target functional element could be resource brokering FE for the downward invocation of resources, and also access control FE for the upward triggering of applications. When the resource is a composite resource, the message routing function will interact with service orchestration FE several times for the sequential request of resources. This function is also responsible for maintaining correlations between requests and responses.

Message transformation

This function is responsible for converting a service request enclosed in an API to common message structure when service request is sent towards a target resource, and converting a common message structure back to the API used by the application when service request is sent towards a target application.

Message exception handling

When message handling errors occur, the message exception handling function tries to resolve them by interacting with the resource registry FE, context management FE and the policy decision FE. The message exception handling function sends back an exception message to the application that owns the error, and asks the resource registry FE to update the state of the resources.

Service logic parsing and processing

The application sends the request to NGN-SIDE. This request could be composed of two parts: the normal resource request and the description of the application's service logic. When the resource request includes the service logic of the application, the service logic parsing and processing function is responsible for:

– parsing the service logic of the application which sends the request for resource invocation;

– processing the request in terms of fetching the status information of the resources requested in the service logic from the context management FE and handling the corresponding procedures according to the acquired status information.

When the resources used in the application service logic are concrete resources (i.e., resources provided by the resource provider and directly exposed by NGN-SIDE) and applications indicate explicitly the resource to be triggered, the service logic parsing and processing function checks the availability of these resources according to the received resource status information. If any of the concrete resources is not available, this function will end the service logic execution and send a failure notification to the application. If all the concrete resources used in the service logic are available, the service logic execution continues.

When one of the resources used in the service logic is an abstract resource (i.e., integrated and exposed by NGN-SIDE and could be triggered as one of multiple concrete resources with similar function provided by different resource providers) and NGN-SIDE chooses one of these resources to be triggered, the service logic parsing and processing function checks the availability of these resources according to the received resource status information. When none of the concrete resources which belong to the same abstract resource is available, this function will end the service logic execution and send a failure notification to the application. When at least one or more concrete resources which belong to the same abstract resource are available and other concrete resources used in the service logic other than this abstract resource are available, the service logic execution continues.

7.2.6 Service orchestration FE

The service orchestration FE enables a collaboration of multiple services (resources) based on their interaction with one another at the message level. This FE consists of two functions and one data source: service composition, composite service execution and service logic repository.

Figure 7-10 shows a functional view of the service orchestration FE.

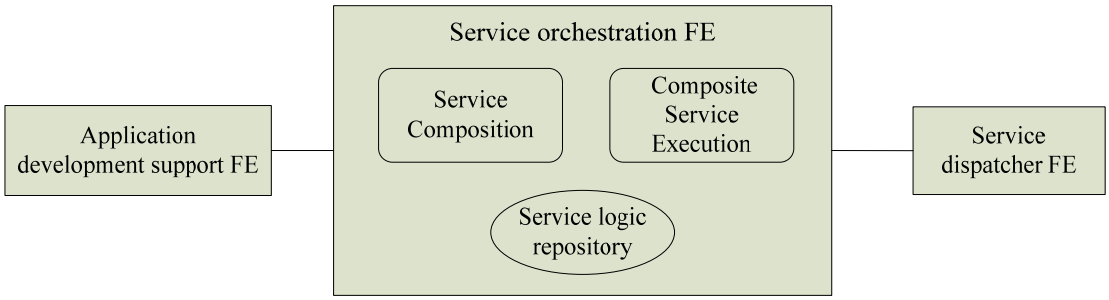


Figure 7-10 – Service orchestration FE functional view

The service composition generates an orchestrated service logic that combines existing service resources to define high level functionalities. The orchestrated service logic describes interactions between multiple services (resources) by exchanging messages, branching logics, and invocation sequences. These orchestrated services (resources) are created using by related orchestrate capabilities and languages such as BPEL and POJO. The service composition provides the ability to expose an orchestrated service (resources) using a specific protocol or interface (REST, SOAP, etc.).

The composite service execution is conducted by parsing and running the service logic. During the composite service execution, sequential request of resources are forwarded to service dispatcher FE.

Composite resource descriptions are stored as service logic. The application development support FE use composite resources in the developing time.

7.2.7 Resource routing FE

For the purpose of interaction with other NGN-SIDEs, the resource routing FE manages and provides resource routing information to locate the resources in other NGN-SIDEs.

Figure 7-11 shows the functional view of the resource routing FE.

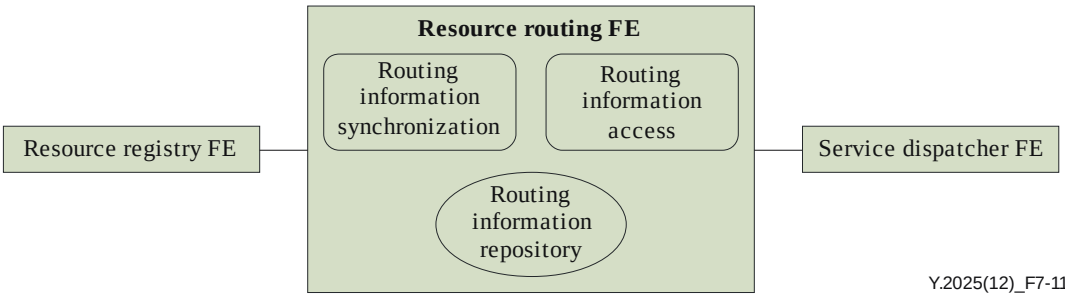


Figure 7-11 – Resource routing FE functional view

The resource routing FE includes the following sub-functions:

– Routing information synchronization

– Routing information access

– Routing information repository.

The routing information repository function stores the resource routing information of the resources which are registered in other NGN-SIDEs.

When a resource is registered to one of the NGN-SIDE, the routing information synchronization function synchronizes the resource routing related information to the resource routing FE in other NGN-SIDEs. For details, refer to Annex A.

The routing information access function interacts with the service dispatcher FE to accomplish the resource request routing function between NGN-SIDEs.

7.3 NGN-SIDE service delivery management functional group related functional entities

7.3.1 Content management FE

The content management FE manages the content, including content storage, content update and content deletion functions.

Figure 7-12 shows a functional view of the content management FE.

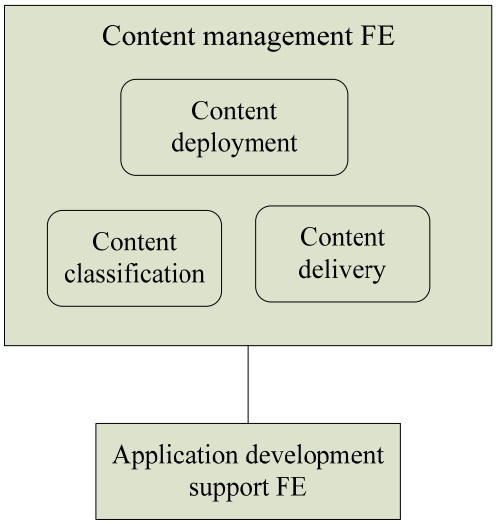


Figure 7-12 – Content management FE functional view

The content classification function extracts the information of the content resource and classifies content resources into different groups. Content resource information includes the content size, type, publisher, location, etc.

The content delivery function performs content delivery to different NGN-SIDE users' environments, such as content for specific applications, content for specific end user equipment, content for specific end user locations, etc.

The content deployment function dispatches content to content storage resources in order to expose content to applications.

7.3.2 Policy management FE

Figure 7-13 shows a functional view of the policy management FE.

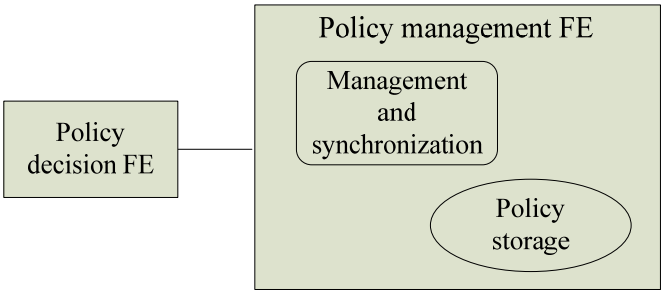


Figure 7-13 – Policy management FE functional view

The policy management FE provides management of policies used for access, provisioning, logging and management of resources.

The policy management FE is responsible for providing a central place for the network provider to manage the rules and policies associated with APIs and services that are exposing through NGN-SIDE. This component will govern and control the lifecycle of the rules/policies. The administrator and provider portal will interface with this FE to provide the following functionalities.

- Create/modify/delete/query rules/policies
- Activate and deactivate rules/policies
- View the status of the rules/policies, which includes:
  a) Current deployed version
  b) APIs or services being used
  c) Display metrics about the API/service performance
- Provision the rules/policies in conjunction with partners and their application as well as APIs, which includes:
  a) Define what rules are to be applied for a given API.
  b) Define a default context that is to be associated with a rule/policy for an given API.
  c) Define a partner's application policy/rule context for a given rule which is to be used when the application invokes that API.
  d) Define how a rule/policy action is to behave for a given API.
- The policy storage in this FE stores these resource related policies, which include SLAs between NGN-SIDE users and NGN-SIDE resources and end user preferences. These policies could be used by policy decision FE to select an appropriate resource for the resource request from applications.

7.3.3 Charging FE

The charging FE provides charging information collection and synchronization function as well as local storage of the call detail records (CDRs).

Figure 7-14 shows a functional view of the charging FE.

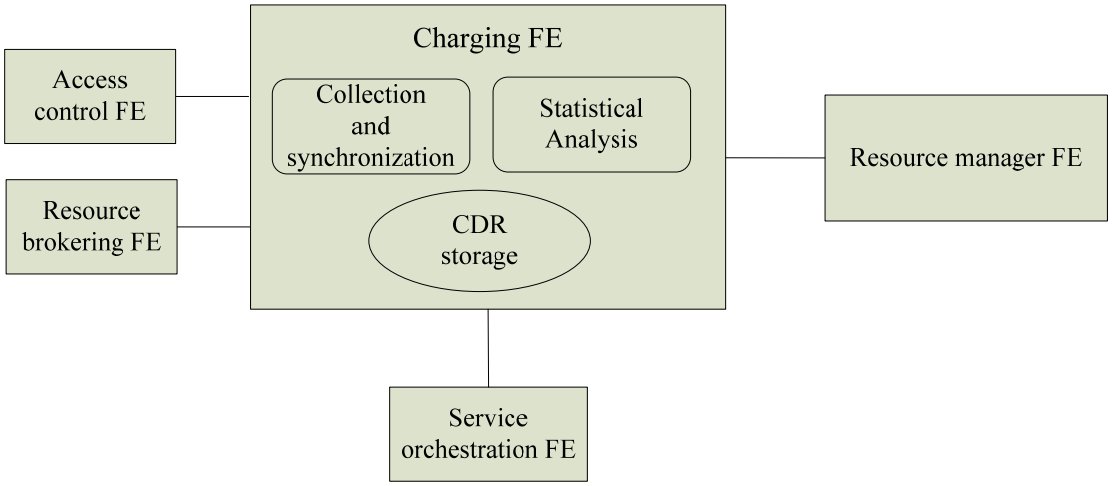


Figure 7-14 – Charging FE functional view

The collection and synchronization function receives the charging information from other FEs in NGN-SIDE, such as the access control FE, service orchestration FE and the resource brokering FE.

When it is required to generate CDRs, charging FE generates CDRs according to the received charging information and stores the CDRs in the CDR storage which are synchronized with the billing domain and backend systems.

The statistical analysis function analyses the resource usage according to the CDRs. Statistical analysis supports multiple dimensions (e.g., resource type, application, developer, time dimension). The charging FE supports event-based online/offline charging mechanisms as specified in [ITU-T Y.2233] and it could be mapped to charging collection function, online charging function and charging gateway function in the functional architecture of charging and accounting in NGN. The charging FE enables SP to implement differentiated charging according to the quality of service (QoS) of the resources as defined in SLA.

When it is required to charge NGN-SIDE users (i.e., application provider), the access control FE triggers a charging event to send charging-related information to the charging FE.

When it is required to support charging for NGN-SIDE resource providers (i.e., short message service (SMS) provider), the resource brokering FE triggers a charging event to send charging-related information (e.g., resource usage information) to the charging FE.

When composite resource whose service logic is stored in service orchestration FE is executed, the service orchestration FE triggers a charging-event to transfer charging related information to the charging FE.

In case of charging for composite resources, the charging FE generates a transaction identifier against this composite resource. Service orchestration FE includes this transaction identifier in the charging information. The service orchestration FE notifies the end of a service transaction to the charging FE.

When it is required to charge for the consumption of resources including computing, connectivity and storage, the resource manager FE sends charging information related to the resource consumption.

7.3.4 Management of role related information FE

Figure 7-15 shows a functional view of the management of role-related information FE.

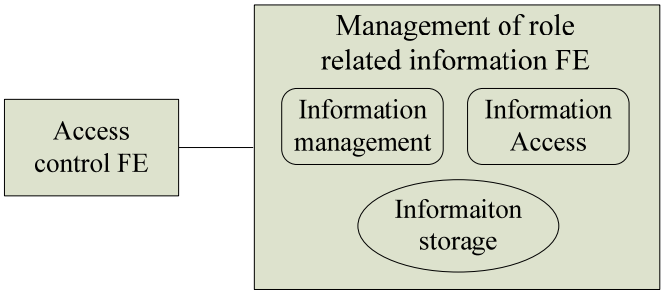


Figure 7-15 – Management of role related information FE functional view

The management of the role-related information FE provides management, storage and access control for all role related information in the NGN-SIDE ecosystems.

The role-related information includes basic information for all roles (e.g., NGN-SIDE user-resource provider relationship), role-specific information (e.g., preference profiles) and subscription-related information (e.g., end user-application provider subscription relationship information for in-house applications). The information storage is provided in the management of role-related information FE to store all this information.

The management operations for the NGN-SIDE provider includes add, modify and delete operations of this role-related information, which are implemented by the information management function. When an end user subscribes a new application, the subscription relationship will be added by the information management function into the information storage, and the end user's related preference profiles are stored in the information storage.

The information access function provides role-related information and subscription to the access control FE for authentication and authorization purposes.

The information management function manages the subscription and preferences of different end users with a particular application. The access control FE checks the subscription of an end user with a specific application by interacting with the information access function.

7.3.5 Application provisioning FE

The application provisioning FE provides mechanisms to deploy applications.

Figure 7-16 shows a functional view of the application provisioning FE.

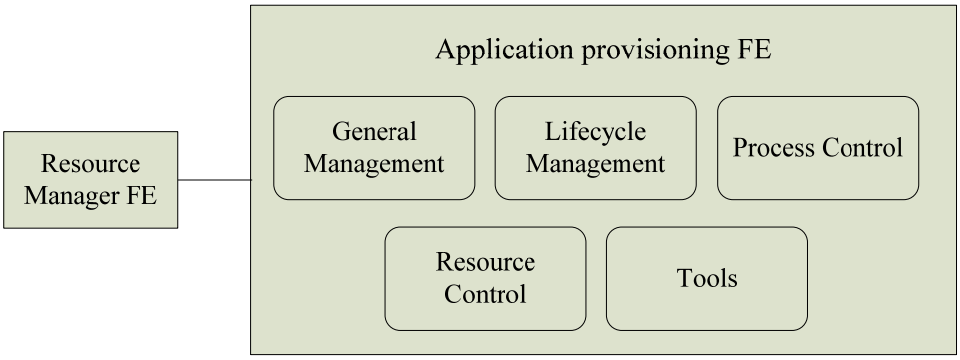


Figure 7-16 – Application provisioning FE functional view

Application provisioning includes the following functions:

– the general management function: provides the general management function of application deployment, such as data management, statistical analysis and so on;

– the lifecycle management function: manages the status of applications in the application deployment process, delivering the start and stop instructions of application deployment and collecting the results;

– the process control function: controls the application deployment process, such as requests number, delay time and so on;

– the resource control function: controls and records dispatch of the resources;

– tools function: includes format conversion tool, which converts the deployment instruction into a suitable format, so as to fit for different running environment; application deployment templates, which could fulfil the automatic deployment and application uninstalling tool. During application uninstall process, uninstalling tool queries the database and other information, then packages operational data of the application into readable structured data.

7.3.6 Context management FE

Figure 7-17 shows a functional view of the context management FE.

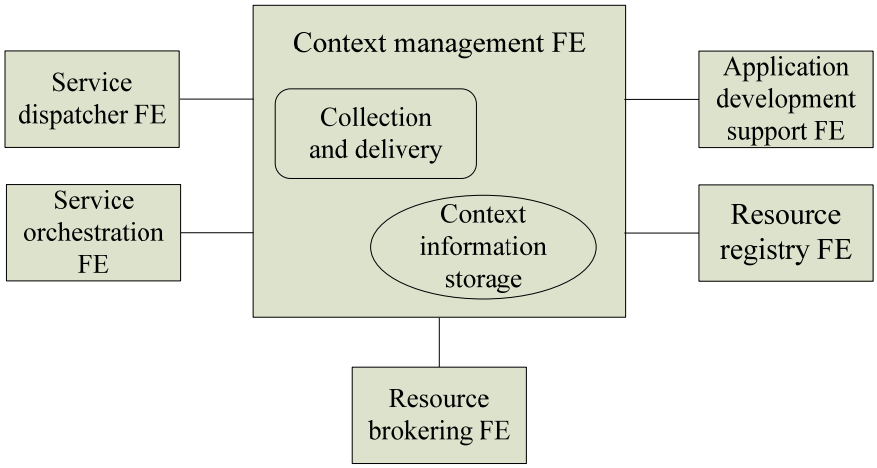


Figure 7-17 – Context management FE functional view

The context management FE collects and aggregates context information related to different context sources, and exposes the context information to other NGN-SIDE FEs according to the policies.

These context information includes the service context, such as service availability, service QoS, service performance; the end user context, such as the end user's identity, end user's presence, end user's location, end user's preferences and end user's social status; the device context, such as device status (device's presence, device's capabilities), device's runtime parameters (CPU, memory, load, performance, etc.); the network context, such as network conditions (e.g., bandwidth, traffic, topology, etc.) and network performance.

This FE provides a collection and delivery function for collecting and managing the different kinds of context information, also the function for exposing that information to other FEs on the NGN-SIDE; and provides a storage function for storing that information.

The resource registry FE synchronizes resource information with the context management FE. When the resource is registered, the context management FE starts to collect the resource context information.

The service dispatcher FE and the service orchestration FE interact with the context management FE to achieve the context information, such as the end user context, the device context, the service context, etc. The context management FE returns the context information of services to the service parsing and processing function of service dispatcher FE according to the received the request.

The resource brokering FE interacts with the context management FE to achieve the context information, such as the resource related context, the network context, etc., to select an appropriate resource for the application.

The application development support FE could use the context information in the context management FE to develop context-based applications.

Both at the beginning and in the middle of an application execution, the context information could be achieved from this FE to customize, personalize, and adapt the application execution according to the actual situation.

7.4 NGN-SIDE adaptation functional group related functional entities

7.4.1 Resource brokering FE

The resource brokering FE provides the interaction between applications and resources for the downward invocation of resources and the upward triggering of applications. It also interacts with the NGN-SIDE integration layer FEs (e.g., policy decision FE) to enforce the resource-related policies such as scheduling of resources [ITU-T Y.2240].

Figure 7-18 shows the functional view of the resource brokering FE.

Resource brokering FE includes the following sub-functions:

– Resource invocation
– Application invocation
– Resource scheduler
– Application scheduler.

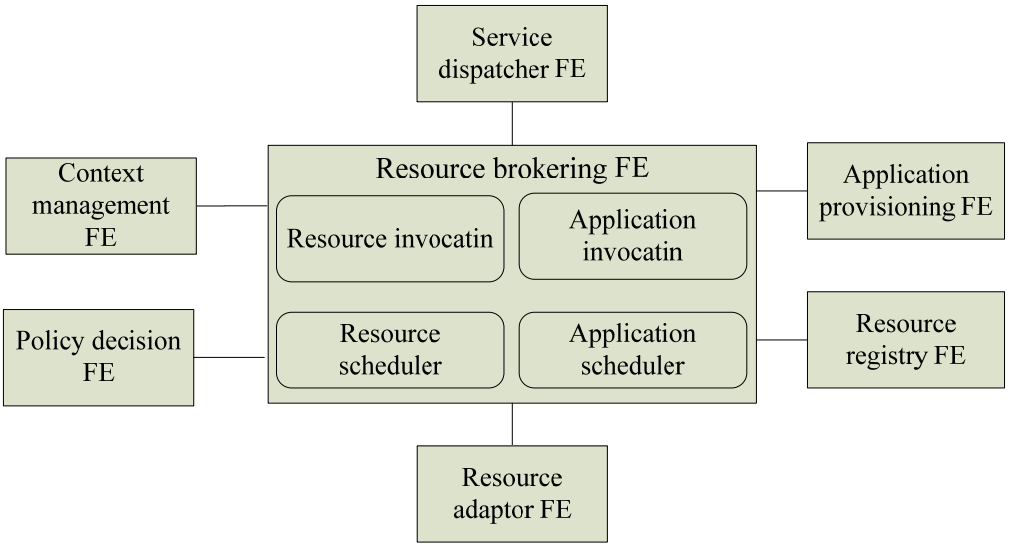


Figure 7-18 – Resource brokering FE functional view

When the resource brokering FE receives a request with a designated resource address (or resource identifier), resource invocation handler parses the request to identify the target resource. The resource invocation handler interacts with the policy decision FE to analyse the related service level agreements (SLAs) and other related policies between NGN-SIDE provider and resource provider. Finally, a request is sent to the corresponding resource adaptor as scheduled by the resource scheduler.

When the resource brokering FE receives a request with a designated application address (or application identifier), the application invocation handler parses and authorizes the service request to identify the target application. The application invocation handler interacts with the policy decision FE to analyse the related service level agreements (SLAs) and other related policies between NGN-SIDE provider and application provider. Finally, a request is sent to the corresponding application as scheduled by the application scheduler.

During runtime, if resource information changes, the resource registry FE notifies the resource brokering FE. The resource brokering FE reflects the corresponding changes in resource invocation and resource scheduler functions.

The resource scheduler is provisioned resource information during resource registration and maintains a list of registered resources with the resource metrics and make available the list of resources based on certain criteria for a particular service. The resource scheduler periodically checks the connectivity and QoS of resource.

The application scheduler is provisioned with application information during application provisioning and maintains application information so that resource requests can be sent to the application when a request is originated from a resource.

7.4.2 Resource adaptor FE

The resource adaptor FE provides mechanisms that allow applications to uniformly access resources by hiding the details of the resources. The resources offered by NGN/non-NGN may vary according to the underlying resource technologies. This makes it difficult for application developers to develop new applications based on direct usage of such diverse and complex underlying resources [ITU-T Y.2240]. To address this, the resource adaptor FE prepares an abstract interface for various resources offered by NGN/non-NGN (e.g., USN, telecom network, broadcasting network, and contents network). The resource adaptor FE performs two types of adaptation: control plan level and data plan level.

The resource adaptor FE receives requests towards resources, and then translates the request message into protocols/APIs according to the target resource protocol/API.

The resource adaptor FE receives requests towards applications, translates the request message into NGN-SIDE internal common messaging format.

When the resource adaptor FE receives a request from the resource registry FE, the resource adaptor FE sets up a corresponding data/control adaptation function for the resource being registered according to the request.

Figure 7-19 shows the functional view of the resource adaptor FE, which includes:

– Control adaptation function
– Data adaptation function
– Resource access function
– Instance management function
– Resource description repository.

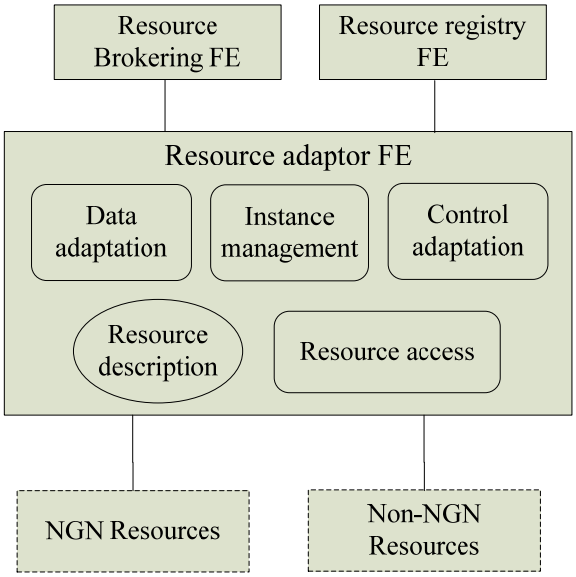


Figure 7-19 – resource adaptor FE functional view

The control adaptation function performs control protocol/API conversions and matches protocol/API request messages between applications and resources.

The data adaptation function performs data adaptation functions to enable applications and resources to deliver data in a heterogeneous environment (e.g., USN data format, and Internet data format).

The resource access function provides an abstract interface to resources in NGN/non-NGN with data adaptation and control adaptation. When a resource is requested through the abstract interface, the resource adaptor FE identifies data type and protocols of the target resource and performs appropriate adaptations for uniform access to various resources.

The instance management function manages all instances of the resource adaptor for a particular resource. It can initiate and remove instances of the resource adaptor.

The resource description repository stores resource description of each resource being adapted. The resource description information includes resource identifier, resource adaptor instance and resource access methods.

8 NGN-SIDE reference points

NGN-SIDE reference points, shown in Figure 7-1 are distinguished into:

– NGN-SIDE internal reference points, related to interactions between NGN-SIDE FEs;

– NGN-SIDE external reference points, related to interactions between NGN-SIDE FEs and functions external to NGN-SIDE.

8.1 NGN-SIDE internal reference points

The following provides the description of the NGN-SIDE internal reference points:

$I_{TE\text{-}ADS}$ Reference point between the testing environment FE and the application development support FE.

The application development support FE can achieve online/offline test tool, test environment and test result through this reference point and make them available to the application developers.

I_{SO-ADS} Reference point between the service orchestration FE and the application development support FE.

The application developer can store their application as resource in service orchestration FE through this reference point, also can achieve the resource kept in service orchestration FE for their own use.

$I_{ADS-CXM}$ Reference point between the application development support FE and the context management FE.

The application development support FE can achieve context information through this reference point.

$I_{ADS-CNM}$ Reference point between the application development support FE and the content management FE.

The application development support FE can achieve content information through this reference point.

I_{ADS-RR} Reference point between the resource repository FE and the application development support FE.

The application development support FE can achieve development interface information of the resources through this reference point.

I_{RR-RE} Reference point between the resource repository FE and the resource registry FE.

The resource repository FE can achieve resource description information from the resource registry FE through this reference point.

I_{AC-SD} Reference point between the access control FE and the service dispatcher FE.

With this reference point, access control FE sends request to service dispatcher FE for further processing. Additionally, service dispatcher FE uses this reference point to transfer request originated from underlying resource (e.g., SMS) towards application.

I_{SD-SO} Reference point between the service orchestration FE and the service dispatcher FE.

With this reference point, service dispatcher FE invokes composite resource in service orchestration FE.

I_{SD-RR} Reference point between the resource registry FE and the service dispatcher FE.

With this reference point, service dispatcher FE interrogates the availability of the requested resource in resource registry FE.

I_{SD-PD} Reference point between the policy decision FE and the service dispatcher FE.

With this reference point, service dispatcher FE interacts with policy decision FE which determines the policies and sends the outcome to the service dispatcher FE.

I_{PD-RM} Reference point between the resource manager FE and the policy decision FE.

With this reference point, resource manager FE consults policy decision FE to allocate resources required for in-house application deployment.

I_{AC-CH} Reference point between the access control FE and the charging FE.

The access control FE can provide charging information through this reference point.

I_{AC-MOR} Reference point between the access control FE and the management of role related information FE.

The access control FE can achieve role related information through this reference point.

I_{SO-CH} Reference point between the service orchestration FE and the charging FE.

The service orchestration FE can provide charging information through this reference point.

I_{PD-PM} Reference point between the policy decision FE and the policy management FE.

The policy decision FE can achieve the policy configured through this reference point.

I_{RM-AP} Reference point between the resource manager FE and the application provisioning FE.

The resource manager FE can achieve the application information configured through this reference point.

I_{RB-CH} Reference point between the resource brokering FE and the charging FE.

The resource brokering FE can provide charging information through this reference point.

I_{RB-CXM} Reference point between the resource brokering FE and the content management FE.

The resource brokering FE can provide context information through this reference point.

I_{RB-AP} Reference point between the resource brokering FE and the application provisioning FE.

The resource brokering FE can achieve application information configured through this reference point.

I_{SO-CXM} Reference point between the service orchestration FE and the context management FE.

The service orchestration FE can achieve context information through this reference point.

I_{SD-CXM} Reference point between the service dispatcher FE and the context management FE.

The service dispatcher FE can achieve context information through this reference point.

I_{RB-PD} Reference point between the resource brokering FE and the policy decision FE.

The policy decision FE can provide related policy configured through this reference point.

I_{RB-RR} Reference point between the resource brokering FE and the resource registry FE.

The resource brokering FE can provide registration information through this reference point.

I_{RB-SD} Reference point between the resource brokering FE and the service dispatch FE.

The service dispatch FE makes use of this reference point to send resource request to resource brokering for the proper selection of the resource.

I_{RB-RA} Reference point between the resource brokering FE and the resource adaptor FE.

The resource adaptor FE can provide adapted resource through this reference point.

I_{RM-CH} Reference point between the resource manager FE and the charging FE.

With this reference point, the resource manager FE provides charging information related to consumption of resources (including computing, connectivity, and storage).

I_{RR-ROU} Reference point between the resource registry FE and the resource routing FE.

With this reference point, the resource registry FE provides resource routing related information to the resource routing FE to synchronize the resource routing information to other NGN-SIDEs.

I_{SD-ROU} Reference point between the service dispatcher FE and the resource routing FE.

The service dispatcher FE achieves resource routing information through this reference point.

8.2 NGN-SIDE external reference points

8.2.1 Reference points related to NGN-SIDE service interfaces

This clause provides the reference points which can be mapped to NGN-SIDE service interfaces identified in [ITU-T Y.2240].

Appendix I provides information about APIs relevant to the NGN-SIDE service interfaces.

$I_{ADF-ADS}$ Reference point between the application development functions and the application development support FE.

With this reference point, developers can access the development environment provided by NGN-SIDE. This reference point can be mapped to the NGN-SIDE ANI or UNI service interfaces.

I_{ADF-TE} Reference point between the application development functions and the testing environment FE.

With this reference point, the developers can access the testing environment provided by NGN-SIDE. This reference point can be mapped to the NGN-SIDE ANI or UNI service interfaces.

I_{APP-AC} Reference point between the applications and the access control FE.

With this reference point, the applications can send resource request with authentication and authorization information to NGN-SIDE. This reference point can be mapped to the NGN-SIDE ANI or UNI service interfaces.

$I_{CPF-CNM}$ Reference point between the content provider functions and the content management FE.

With this reference point, the content provider can make available different types of content as a resource to NGN-SIDE. This reference point can be mapped to the NGN-SIDE SNI service interface identified in [ITU-T Y.2240].

NOTE – Annex A describes interactions between different NGN-SIDEs which can be achieved through the NGN-SIDE NNI service interface identified in [ITU-T Y.2240].

8.2.2 Reference points related to NGN-SIDE resource interfaces

This clause provides the reference points related to NGN-SIDE resource interfaces identified in [ITU-T Y.2240].

I_{RA-NGN} Reference point between the resource adaptor FE and the resources in NGN (see also the relevant information in clauses 9 and 10 of [ITU-T Y.2240]). The resources are provided according to the corresponding request through this reference point.

$I_{RA-NNGN}$ Reference point between the resource adaptor FE and the resources in non-NGN (see also the relevant information in clauses 9 and 10 of [ITU-T Y.2240]).

8.2.3 Reference points to other functions

I_{BD-CH} Reference point between the billing domain and the charging FE.

With this reference point, the billing domain can achieve charging information such as CDR from NGN-SIDE.

$I_{ADM-CNM}$ Reference point between the administrator functions and the content management FE.

With this reference point, the administrator can manage the content.

I_{ADM-AP} Reference point between the administration function and the application provisioning FE.

With this reference point, the administrator can manage the application's lifecycle, deployment status, etc.

9 NGN-SIDE service procedures

9.1 Resource registration procedure

Resources are located and accessed by the application as well as used by the application developer after resources have been registered successfully at the resource registry FE.

Figure 9-1 describes how a resource is registered to NGN-SIDE.

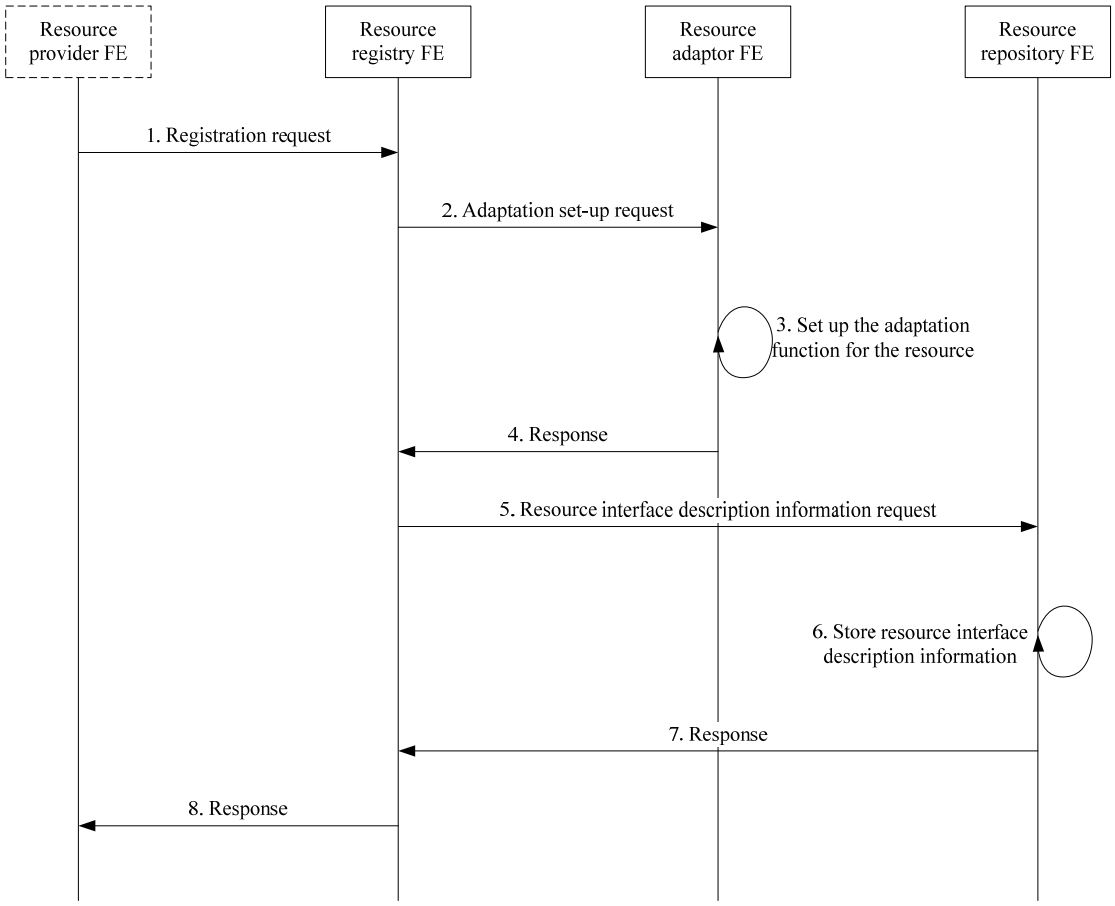


Figure 9-1 – Resource registration procedure

Procedure description:

– Resource provider sends resource registration request to resource registry FE.

– After receiving the request, resource registry FE sends resource adaptation set-up request to adaptor.

– Adaptor sets up the adaptation function corresponding to the resource information.

– Adaptor sends back the response to resource registry FE.

– Resource registry FE sends request which includes resource interface description information to resource repository FE.

– Resource repository FE stores the resource interface description information locally.

– Resource repository FE sends back the response to resource registry FE.

NOTE – Order of step 2 and 5 is implementation dependent and can be executed in any order.

Resource registry FE notifies resource provider the resource registration result.

9.2 Application development procedure

This procedure describes how developers access the application development related functions provided by NGN-SIDE to develop and test applications which using the resources exposed in NGN-SIDE.

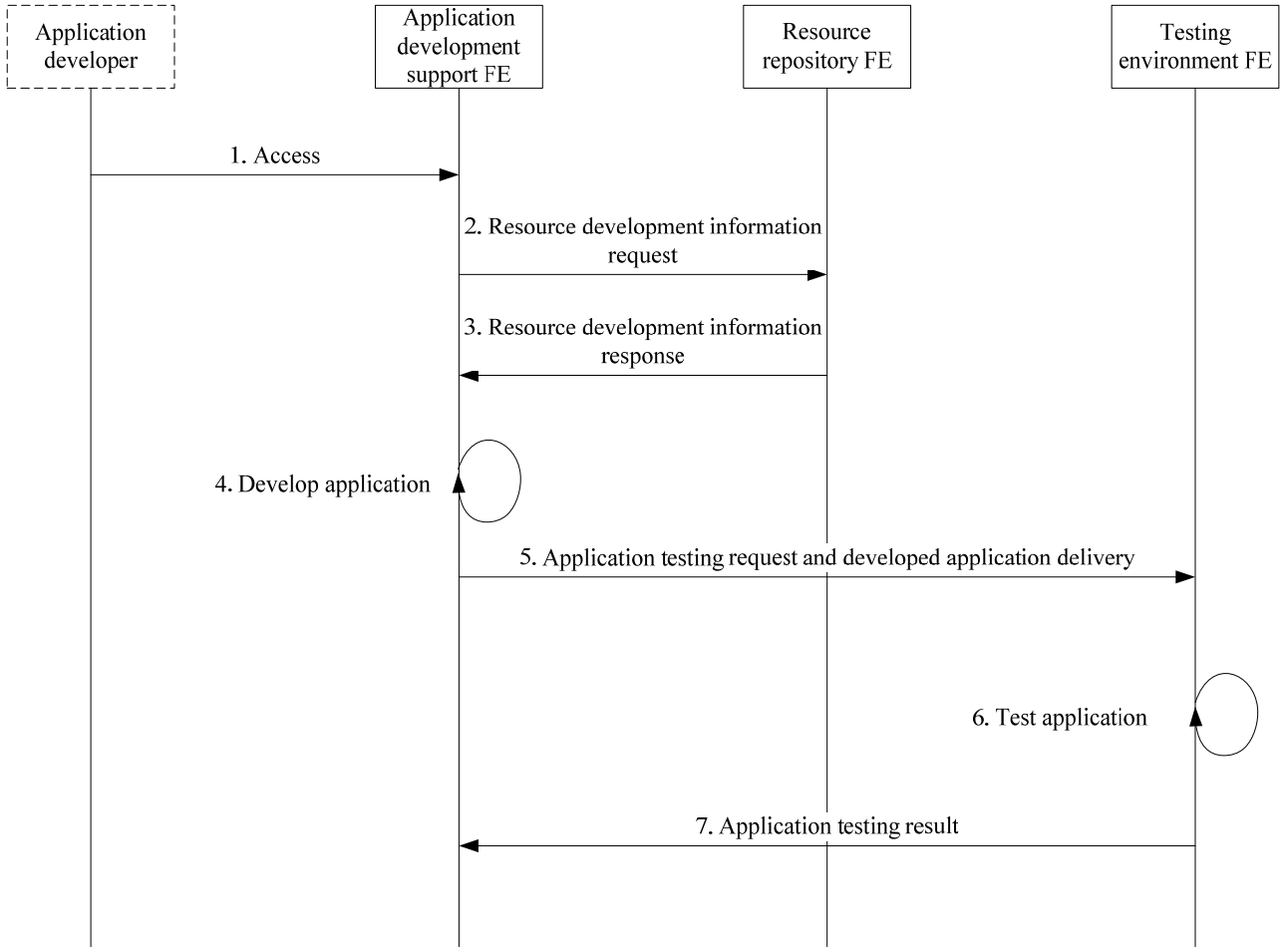


Figure 9-2 – Application development procedure

Application developer accesses the application development support FE to create application.

– The application development support FE supports interaction with the resource repository FE to obtain the development interface information of the resources required in the application development.

– The resource repository FE returns the development interface information of the resources.

– Application developer creates applications using online and/or offline design tools according to the resource development interface information.

– When application is created, application developer requests testing of application and deliver the developed application to the testing environment.

– The testing environment FE performs the testing procedures for the application.

– The testing environment FE returns the application testing result.

9.3 Application provisioning procedure

The application provisioning FE provides lifecycle management for applications in NGN-SIDE such as deployment, activation, deactivation and withdrawal. In those procedures, the application provisioning FE interacts with the administrator and provider portal (i.e., user interface), resource manager FE and resource brokering FE.

9.3.1 Application deployment

The application deployment procedure stores provisioning information of the target application such as a name addresses and requirements in NGN-SIDE. Figure 9-3 depicts how an application is deployed into NGN-SIDE.

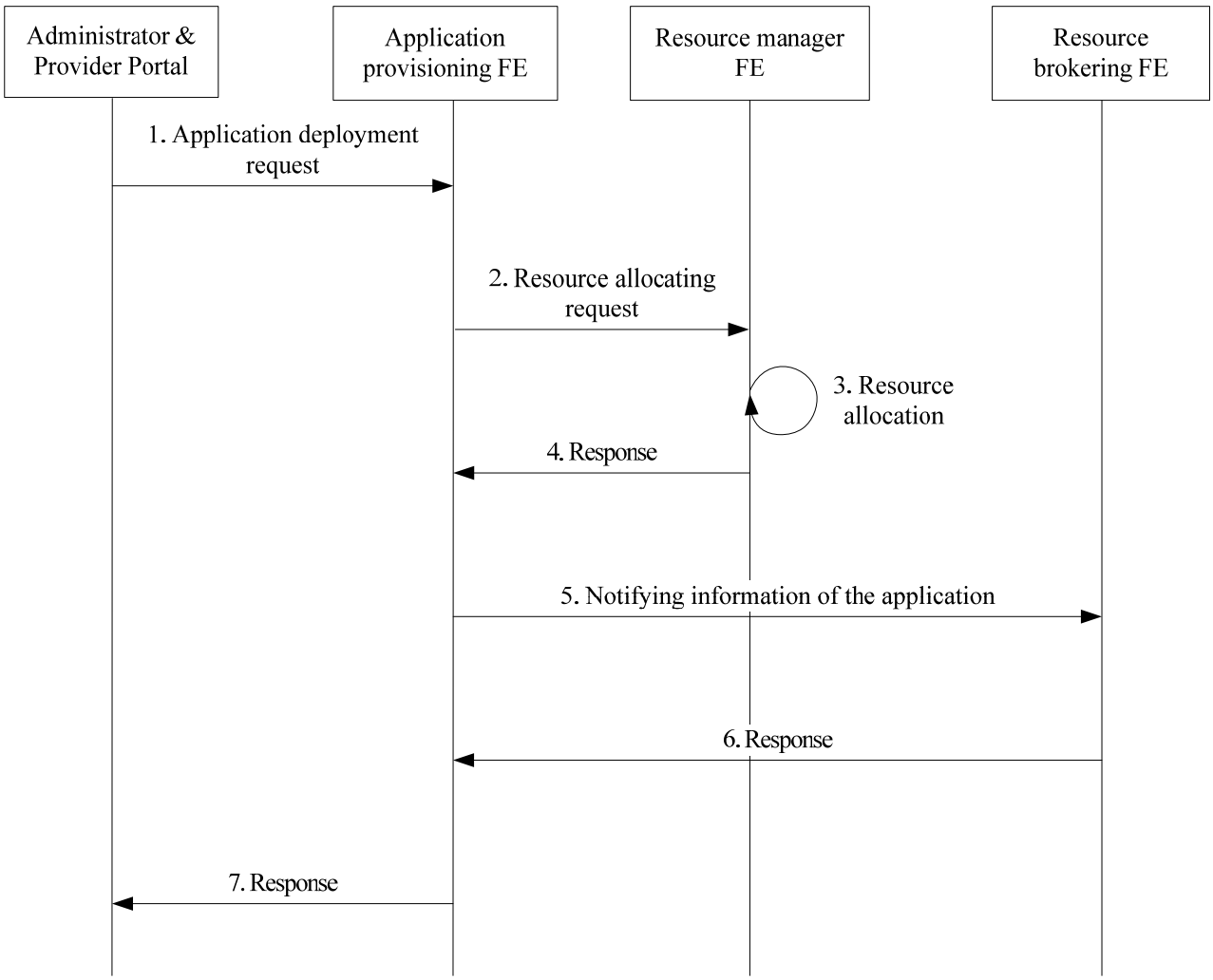


Figure 9-3 – Application deployment processes

– An administrator requests deployment of an application through the provider portal interface. The application related information is inserted into application provisioning FE.

– Application provisioning FE interprets the application related information that contains an application name, an application accessible address, and resource requirements and so on.

– Application provisioning FE sends resource requirements to resource manager FE.

– Resource manager FE allocates resources for the deploying application.

– Resource manager FE sends back the result to application provisioning FE.

– After required resources are allocated for the application, application provisioning FE notifies the information of the application deployment to resource brokering FE.

– Resource brokering FE maintains state of the application (i.e., deployed) and sends back the response to application provisioning FE.
– Application provisioning FE sends back the result of the application deployment to the provider portal.

9.3.2 Application activation/deactivation

The application activation/deactivation procedure switches state information of an application whether it is available or not. Figure 9-4 depicts how an application is activated or deactivated.

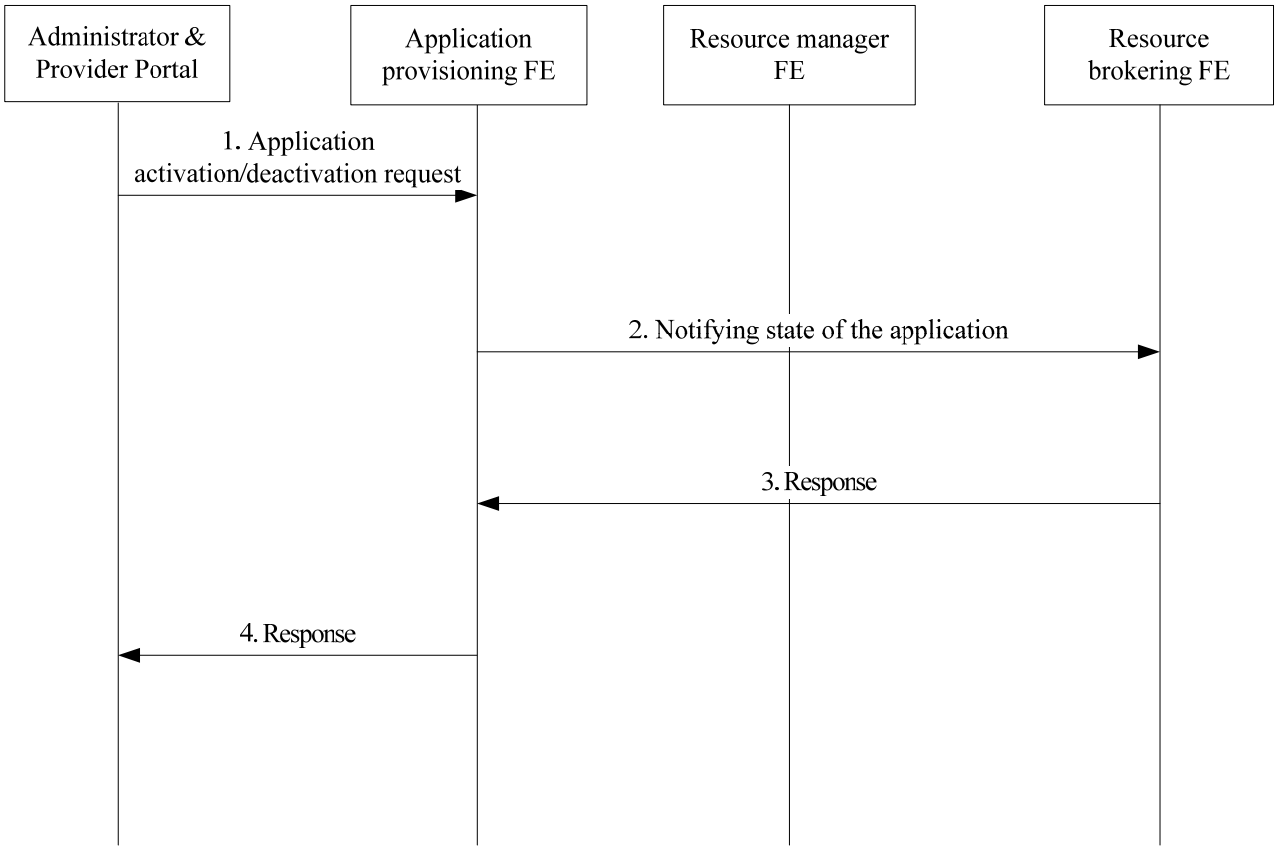


Figure 9-4 – Application activation/deactivation processes

An administrator requests activation or deactivation of an application through the provider portal interface. The target application identifier is inserted into application provisioning FE.

– Application provisioning FE notifies the state of them to resource brokering FE.
– Resource brokering FE updates state of the application (i.e., activated or deactivated) and sends back the response to application provisioning FE.
– Application provisioning FE sends back the result of the application activation or deactivation to provider portal.

9.3.3 Application withdrawal

The application withdrawal procedure eliminates the application provisioning information and related resources from NGN-SIDE. Figure 9-5 depicts how an application is removed.

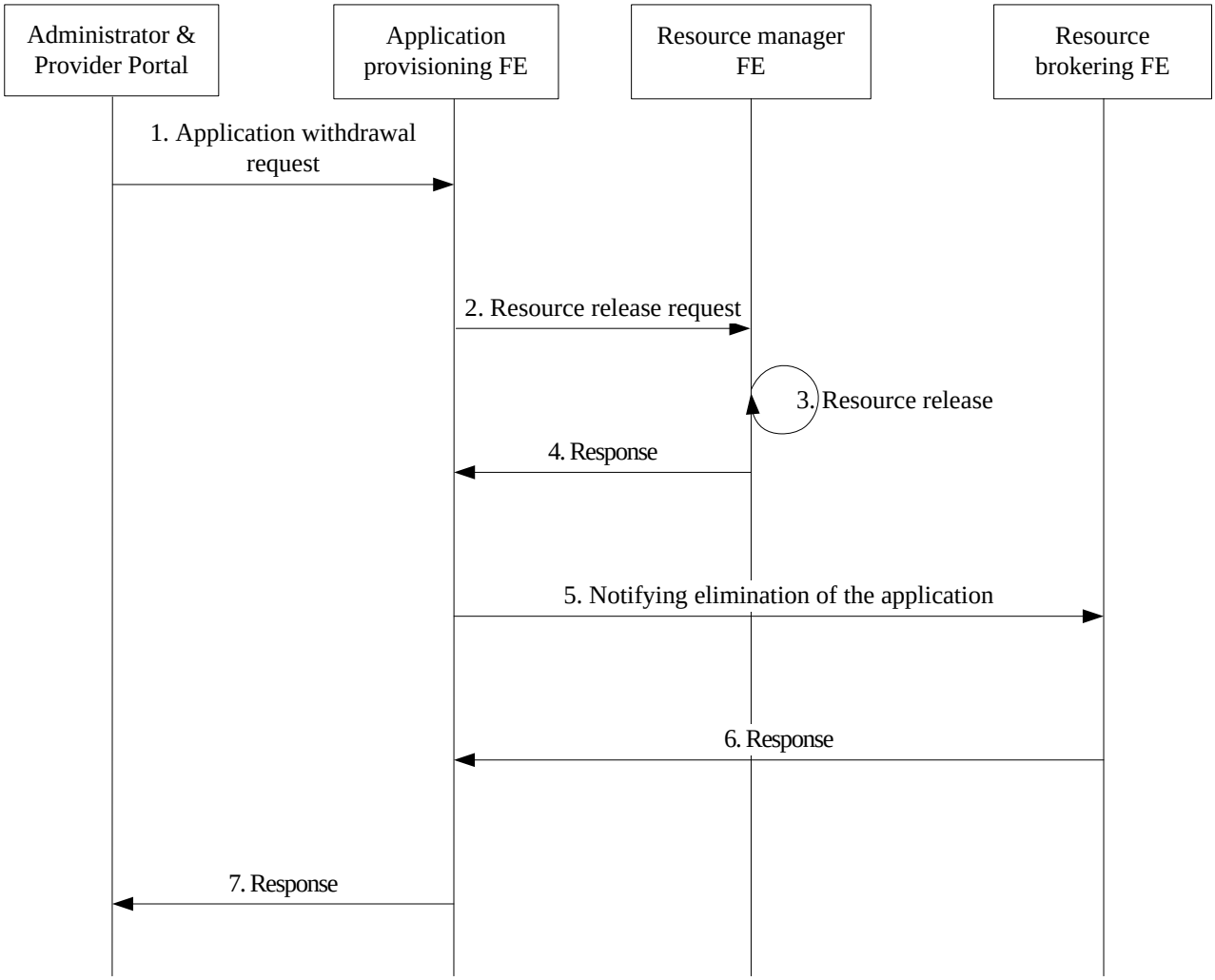


Figure 9-5 – Application withdrawal processes

An administrator requests withdrawal of an application through the provider portal interface. The target application identifier is sent to the application provisioning FE.

– The application provisioning FE sends a withdrawal command with the target application identifier to resource manager FE.
– Resource manager FE removes the all resources allocated to the application.
– Resource manager FE sends back the result to application provisioning FE.
– Application provisioning FE notifies the withdrawal of the application to resource brokering FE.
– Resource brokering FE eliminates the application related information in it and sends back the response to application provisioning FE.
– Application provisioning FE eliminates the application related information and sends back the result of application withdrawal to the provider portal.

9.4 Resource triggering procedure

This procedure describes the service flow to invoke a particular resource. This service flow includes interaction of access control FE, service dispatcher FE, resource registry FE, resource brokering FE and resource adaptor FE.

In the following service flows, application has subscription to a location service enabler to find the user's location.

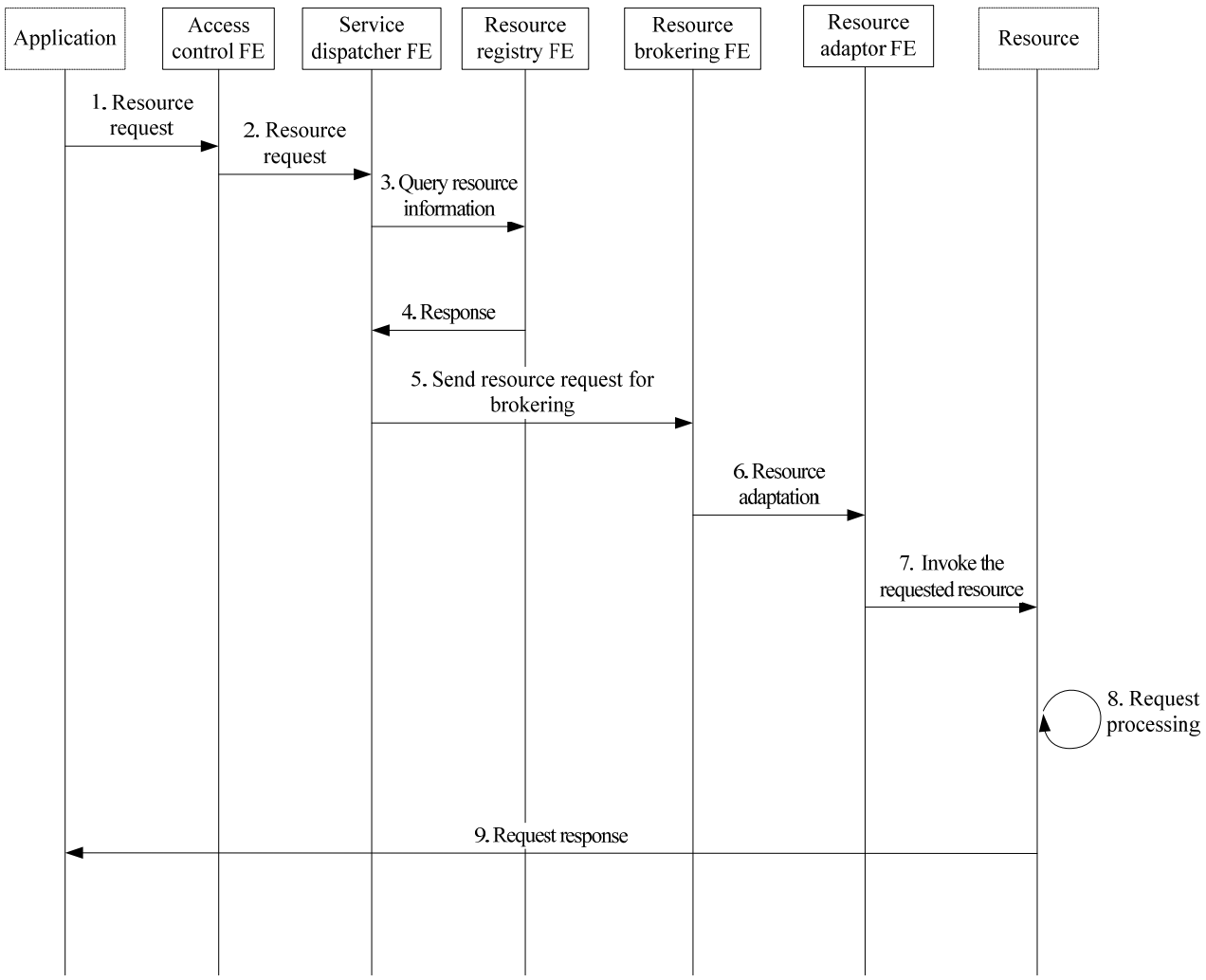


Figure 9-6 – Resource triggering procedure

– Application sends a resource (location) request to get location information of a user.

– Access control FE authenticates and authorizes requesting application.

– Service dispatcher translates the API/protocol carrying the request into a common message structure and enquires the resource registry FE to get the resource information.

– Resource registry returns the resource information to the service dispatcher FE.

– Service dispatcher sends the resource request to resource brokering FE.

– Resource brokering FE selects the resource among one or more candidate resource list and sends the resource request to the corresponding resource adaptor.

– Resource adaptor generates a message according to the API/protocol native to the target resource and encloses content received in common message structure and invokes the target resource.

– The target resource processes the request (get user location).

– Target resource (location service enabler) sends response (carrying user location) towards application via different FEs.

9.5 Context information-based service procedure

In NGN-SIDE, the context information is utilized to select the most appropriate resources for the purpose of services or applications. The context management FE maintains in its own storage many types of context information such as the service context, the end user context, the device context and the network context.

The main function of context information-based service procedure consists of delivery and collection of context information. First, delivery of context information is performed when the context management FE receives context requests from external entities that serve context information-based service. Second, context information is collected by context management FE continuously.

Figure 9-7 depicts how context information-based services are conducted. Context using functional entities include the service dispatcher FE, service orchestration FE and application development support FE. Hereafter, those kinds of functional entities are indicated as context utilizing functional entities.

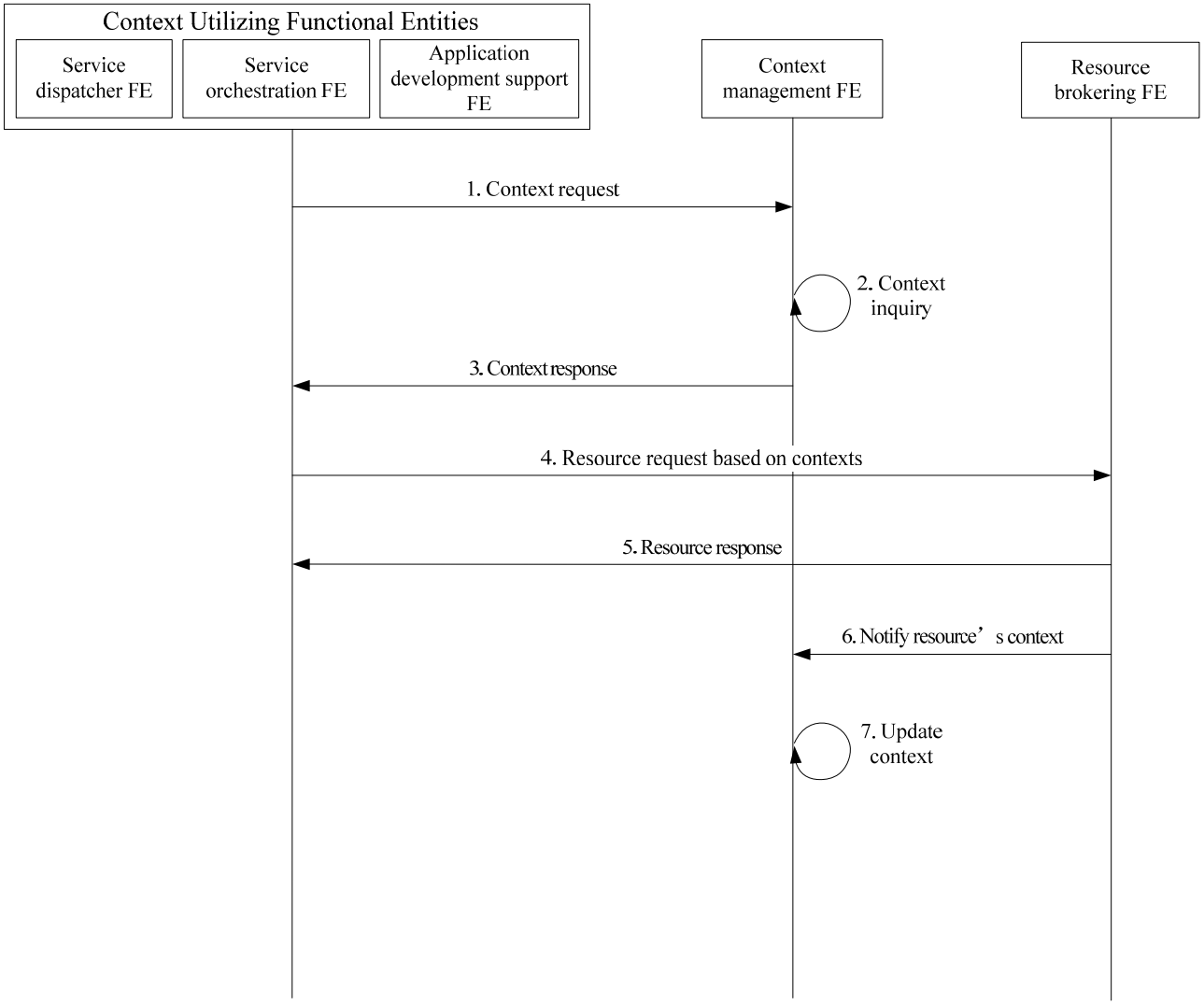


Figure 9-7 – Context information-based service processes

– Context utilizing functional entities request context information to context management FE.

– Context management FE looks up requested context information at context management storage.

– Context management sends back the context information to requested FE.
– Context utilizing functional entities request resources based on context information.
– Resource brokering FE returns resources to requested FE.
– Resource brokering FE notifies context information to context managements FE if they have any change.
– Context management FE updates context information.

9.6 Charging procedure

9.6.1 Charging procedures for NGN-SIDE user

When the access control FE receives a request from an NGN-SIDE user (i.e., application), after authentication and authorization, the access control triggers charging event to send charging related information to the charging FE.

The charging FE collects information on the service provider (SP) consumption of resources and generates call detail records (CDRs). The charging FE enables SP to implement differentiated charging according to the quality of service (QoS) of the resources as defined in SLA.

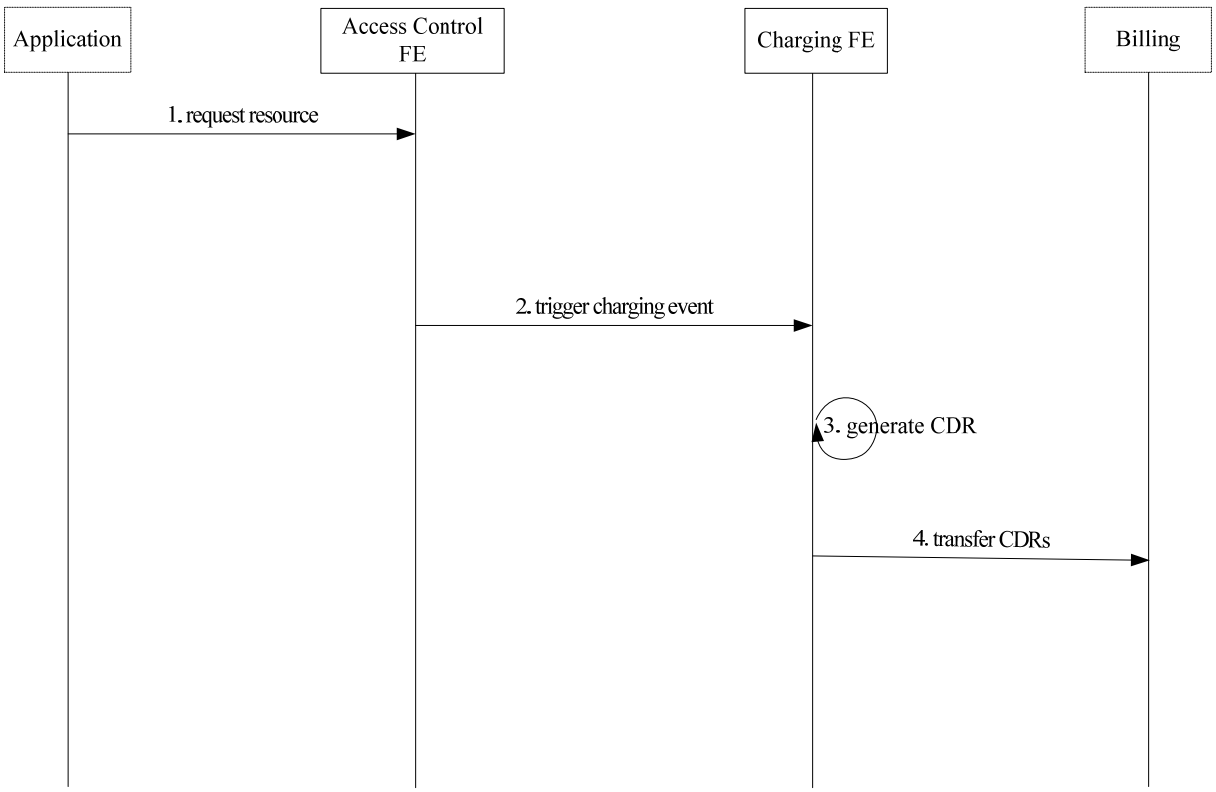


Figure 9-8 – Charging procedure for NGN-SIDE user (application provider)

– The application sends request to invoke resource.
– The access control FE triggers charging event to send charging information related to the resource request.
– The charging FE generates CDR according to received charging information.
– The charging FE sends CDRs to the backend billing system.

9.6.2 Charging procedures for composite resources

When composite resource whose service logic is stored in service orchestration FE is executed, the service orchestration FE triggers a charging event to transfer charging-related information to the charging FE.

The charging FE generates TransactionID against this composite resource. The service orchestration FE is required to always include this TransactionID in the charging information. The service orchestration FE notifies the end of a service transaction to the charging FE.

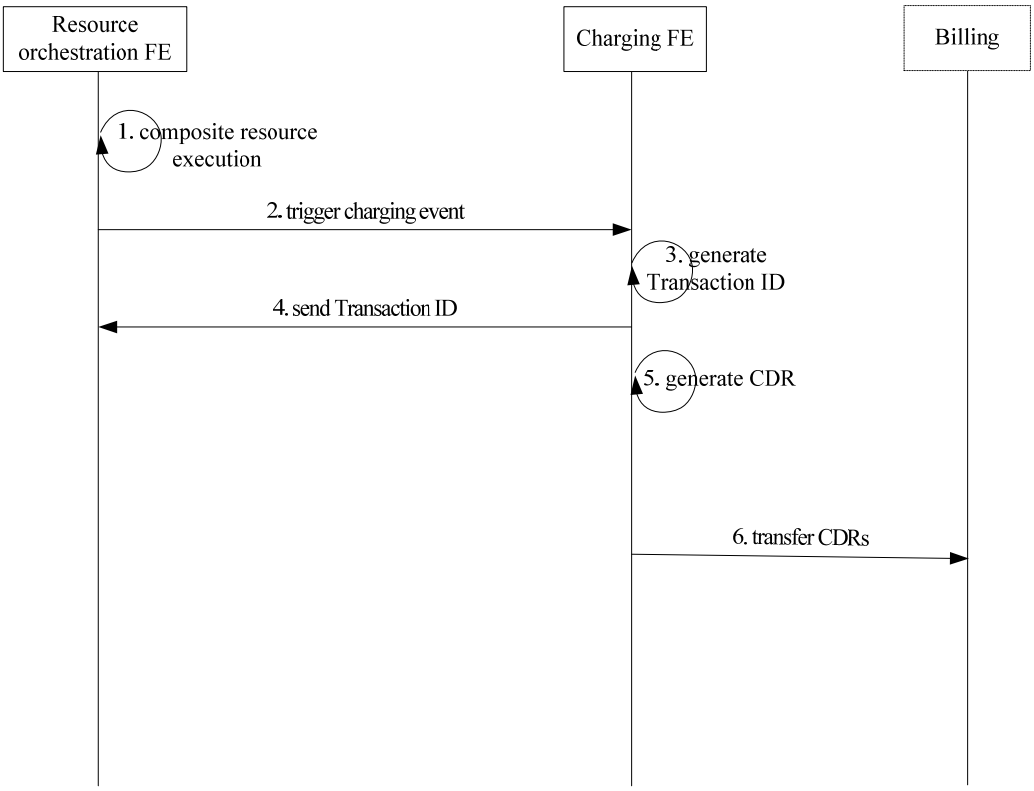


Figure 9-9 – Charging procedure for composite resources

– The resource orchestration FE is executing the composite resource logic.

– The service orchestration FE triggers charging event to send charging related information with the indication about composite resource.

– The charging FE generates service TransactionID.

– The charging FE sends TransactionID to service orchestration FE.

– When composite resource transaction ends, service orchestration notifies the charging FE to close the charging transaction for the composite resource. As a result, the charging FE generates a CDR according to the charging information.

– The charging FE sends CDRs to backend billing system.

9.6.3 Charging procedures for NGN-SIDE resource provider

In NGN-SIDE ecosystem, NGN-SIDE resource providers and NGN-SIDE resource users can have independent charging triggering mechanisms according to the deployment requirements.

This scenario describes the charging procedures for the NGN-SIDE resource provider.

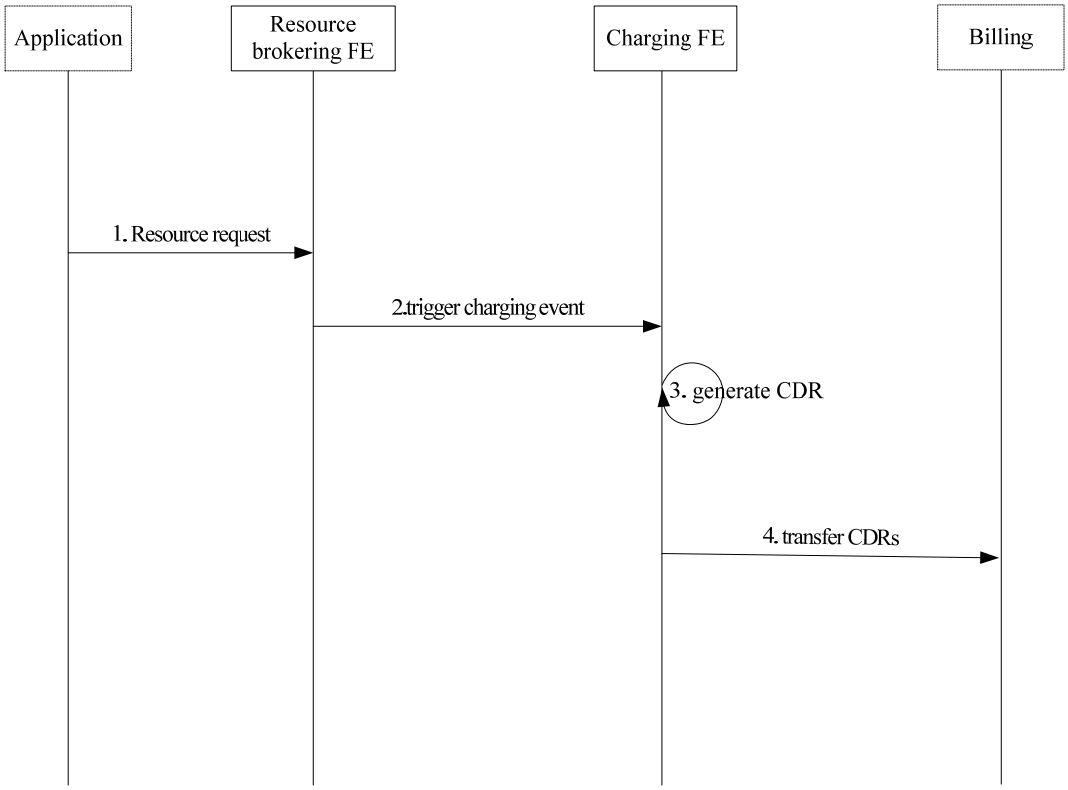


Figure 9-10 – Charging procedure for NGN-SIDE resource provider

– Request from application is sent towards resource brokering FE.

– The resource brokering FE triggers charging event to send charging information related to the invoked resource.

– The charging FE generates CDR according to received charging information.

– The charging FE sends CDRs to backend billing system.

9.7 Application triggering procedure

This procedure provides the service flows related to triggering an application (e.g., vote casting application for counting votes) by a resource (e.g., SMS). This service flow includes interaction of resource adaptor FE, resource brokering FE, service dispatcher FE and access control FE.

In this service flow, the resource submits (messaging service) a message to an application which is deployed to count votes.

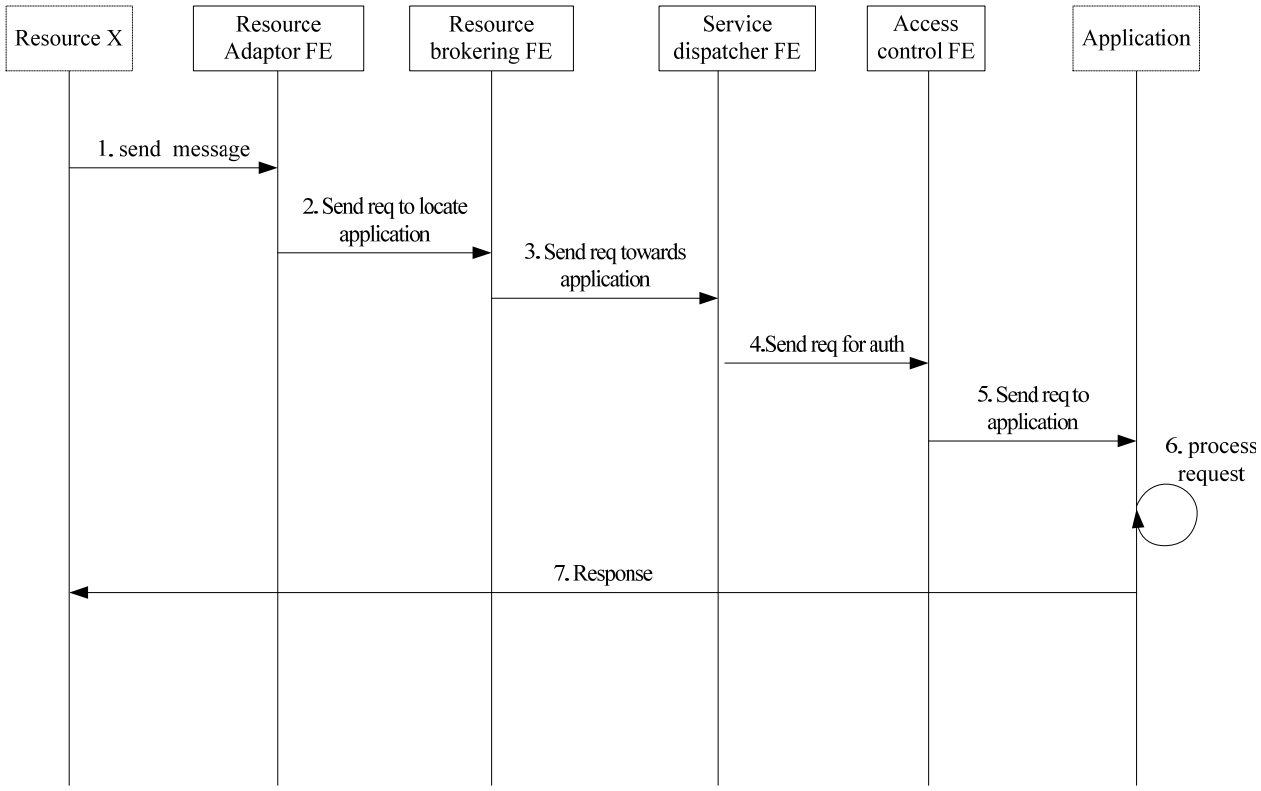


Figure 9-11 – Application triggering procedure

– Resource X sends a message carrying vote information towards an application used for vote counts via resource adaptor FE.

– Resource adaptor corresponding to resource X translates the API/protocol carrying the message and encloses the message content into a common message structure and send the message to resource brokering FE.

– Resource brokering FE locates the corresponding application and sends the request to service dispatcher FE.

– Service dispatcher FE interacts with other relevant functional elements (e.g., policy decision) for policy enforcement. Afterward service dispatcher FE generates message according to the API/protocol native to the application and encloses content received in common message structure. This message is sent to access control FE.

– Access control FE performs the authentication and authorization and delivers the message to the application.

– Application processes the message content to get the vote information and update the vote count accordingly.

– Application sends a message confirmation response to the resource via different FEs.

9.8 Procedure related to accessing resource in another NGN-SIDE

Figure 9-12 describes the scenario in which an application deployed in a CSN node accesses the resource registered in a remote CSN node.

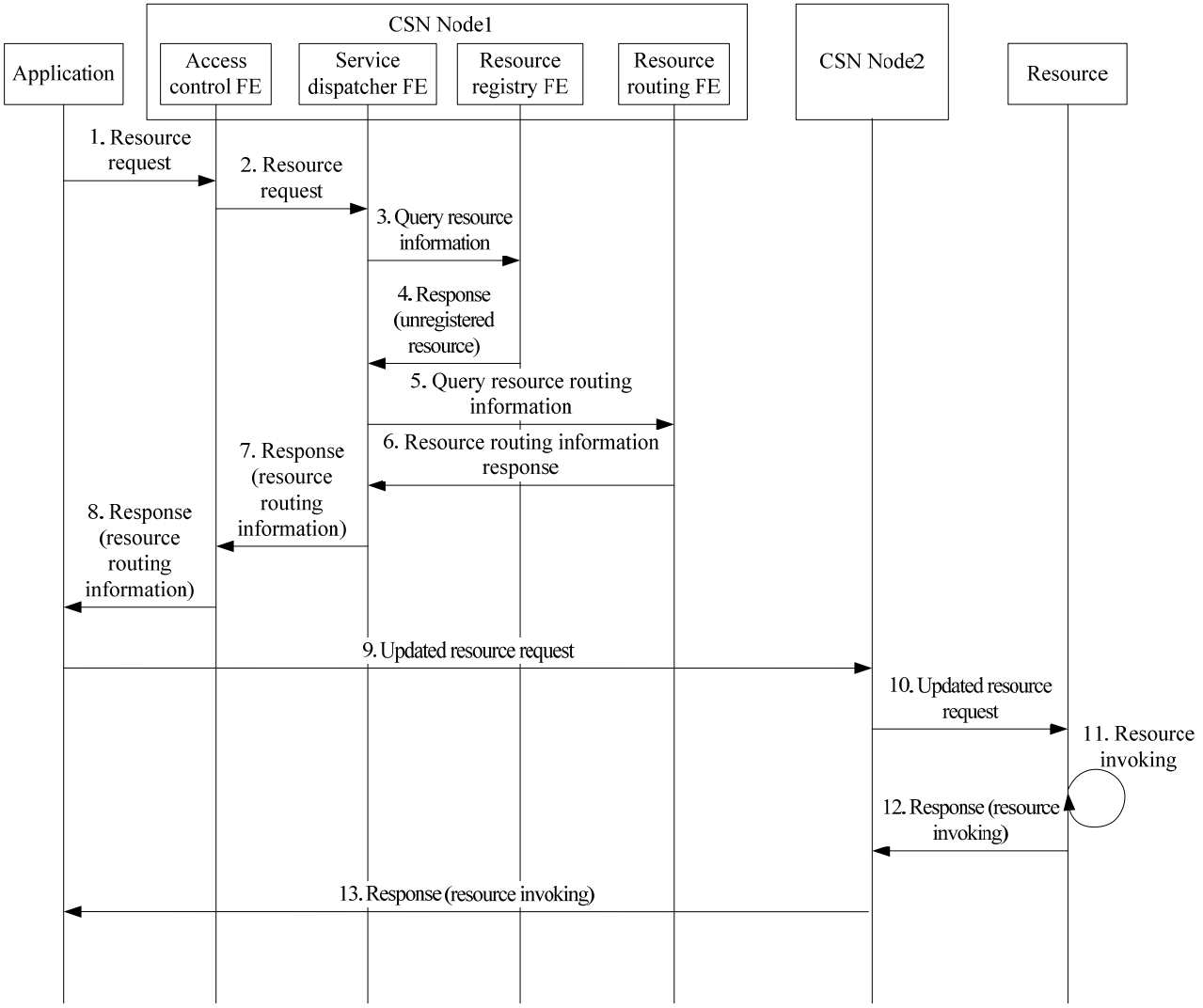


Figure 9-12 – Procedure related to accessing resource in remote CSN-Node

Procedure description:

– Application sends resource request to CSN-Node1.

– Access control FE in CSN-Node1 authenticates the resource request, and transfers it to service dispatcher FE upon successful authentication.

– Service dispatcher FE queries resource information through resource registry FE.

– Resource registry FE returns the query response, which indicates the resource does not register in this CSN-Node.

– Service dispatcher FE queries resource routing information through resource routing FE.

– Resource routing FE returns the routing information of the requested resource.

– Service dispatcher FE transfers the routing information to access control FE.

– Access control FE transfers the routing information to application.

– Application sends updated resource request to CSN-Node2.

– CSN-Node2 sends request to resource.
– Resource is invoked.
– Resource sends invoking response to CSN-Node2.
– CSN-Node2 sends resource invoking response to application.

10 Security considerations

The security requirements for the functional architecture of the NGN-SIDE are addressed by the security requirements identified in clause 8.4 of [ITU-T Y.2240].

Annex A

Interconnection between different NGN-SIDEs

(This annex forms an integral part of this Recommendation.)

NGN-SIDE in a particular NGN domain exposes its resources to the NGN-SIDE users. The NGN-SIDE users of a certain NGN-SIDE can only use the resources registered in that NGN-SIDE. In some circumstances (e.g., population density, geography), an NGN-SIDE provider can deploy distinct NGN-SIDEs in different NGN domains. These NGN-SIDEs can collaborate with each other to enable the users of an NGN-SIDE to access diverse resources available in other NGN-SIDEs.

The interconnection of these NGN-SIDE forms the so called converged service network (CSN). Each NGN-SIDE in the CSN is known as CSN-Node. Figure A.1 shows the overall concept of CSN.

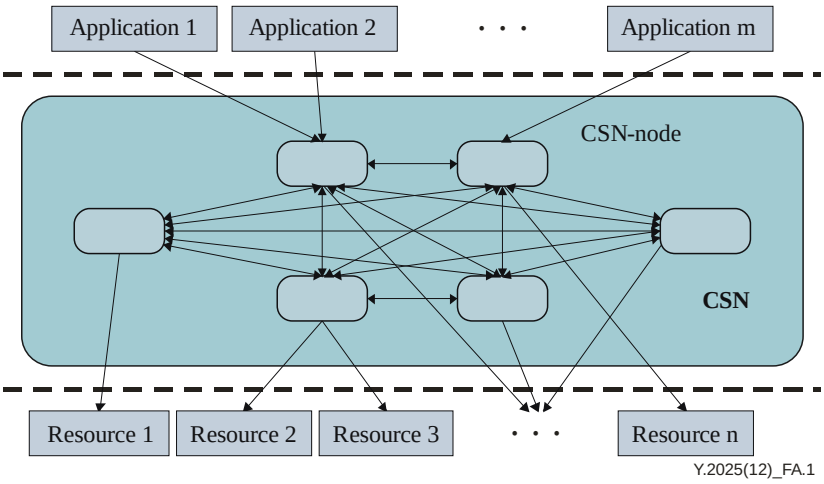


Figure A.1 – CSN conceptual view

As shown in Figure A.1, the CSN-Nodes are interconnected with each other in the CSN. Each CSN-Node manages one or multiple categories of resources.

When resource requests from applications are received by CSN-Nodes, the CSN-Nodes schedule the resources for these applications, that is, select and invoke the requested resources.

Figure A.2 shows the internal structure of a CSN-Node.

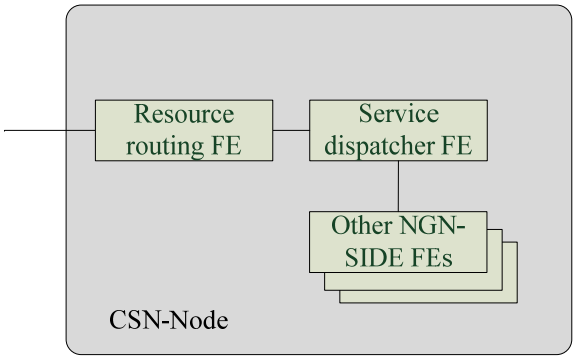


Figure A.2 – CSN-Node internal structure

The service dispatcher FE analyses the received resource request and queries the resource registry FE whether the requested resource is registered in this CSN-Node. When the requested resource is not registered in this CSN-Node, the service dispatcher FE sends the query request to the resource routing FE to acquire the routing information of the requested resource.

When receiving the request, the resource routing FE queries the routing information of the requested resource in its storage and returns the routing information to the service dispatcher FE.

The service dispatcher FE returns the routing information of the requested resource to the application, and the application will request the resource in another CSN-Node according to the routing information. Alternatively, the service dispatcher FE could directly route the resource request to the CSN-Node in which the resource registers.

The CSN-Nodes can also be organized and interconnected in hierarchical mode. Figure A.3 provides the CSN conceptual view in n-tiers hierarchy.

The CSN-Node on the top most tier (e.g., tier n-1) is responsible for the overall routing of resources in a hierarchical manner.

NOTE – Applications accessing a specific CSN-Node can locate resources registered to any CSN node in any tier.

When a CSN-Node on a particular tier (e.g., tier n) identifies that the requested resource is not locally registered but available in underlying CSN-Nodes, it routes the request to the CSN-Node on the lower tier (e.g., tier n+1). The CSN-Node (s) on the lower tier(s) then query(ies) the requested resource tier by tier until the resource is found.

When a CSN-Node on a particular tier (e.g., tier n) identifies that the requested resource is not registered locally, it routes the request to the CSN-Node on the upper tier (e.g., tier n-1) or directly to the CSN-Node in the top most tier to locate the resource.

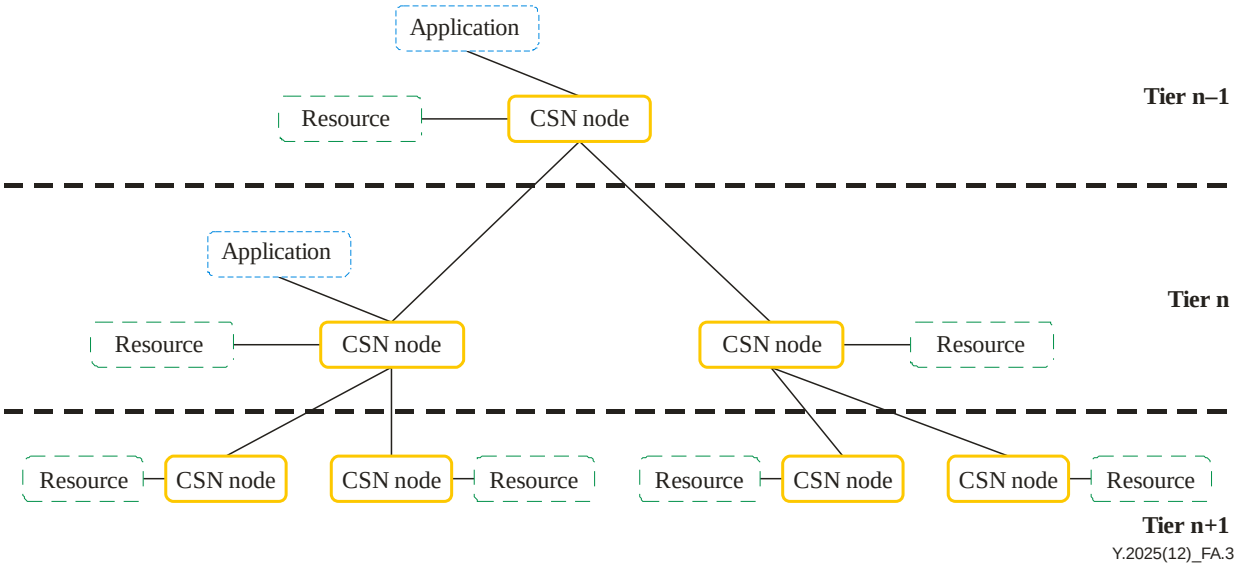


Figure A.3 – CSN conceptual view in hierarchical mode

Appendix I

APIs for the NGN-SIDE functional architecture

(This appendix does not form an integral part of this Recommendation.)

This Recommendation does not intend to define application programming interfaces (APIs) architectural aspects. The exposure of APIs for functional architecture of the NGN-SIDE is achieved according to requirements identified in clause 10.2.2 of [ITU-T Y.2240].

For detailed information on APIs across different service interfaces, refer to the following clauses of [ITU-T Y.2240]:

– 10.2.2 "NGN-SIDE service interfaces"
– 10.2.2.1 "General requirements of NGN-SIDE service interfaces"
– 10.2.2.2 "Service interface requirements across ANI"
– 10.2.2.3 "Service interface requirements across UNI"
– 10.2.2.4 "Service interface requirements across NNI"
– 10.2.2.5 "Service interface requirements across SNI"

Appendix II

Mapping between NGN-SIDE functional entities and NGN functional entities

(This appendix does not form an integral part of this Recommendation.)

The NGN-SIDE architecture is defined in accordance with [ITU-T Y.2012]. Therefore, its functionalities have a corresponding relationship with the NGN architecture. Figure 6-1 shows the location of NGN-SIDE in the NGN functional architecture.

Table II.1 provides the relationships between the functional entities identified in this Recommendation and the functional entities in the NGN architecture [ITU-T Y.2012].

Table II.1 – Mapping between NGN-SIDE functional entities and NGN functional entities

No.	NGN-SIDE functional entity	NGN functional entity	Remarks
1	Content management FE	Content distribution and location control FE	Both control the distribution of content and gather the related information.
2	Service orchestration FE	Application service coordination manager FE	Both handle the interaction of several resources or services.
3	Context management FE	Application support user profile FE	Application support user profile FE in NGN support end user settings which include information related to the capabilities of the end user's terminal devices.
4	Application provisioning FE	Application provisioning FE	Both support lifecycle management of the applications.
5	Resource registry FE, role related information management FE	Identity management functions	Resource registry FE and role related information management FE provide support for IdM mechanisms.
6	Role related information management FE, policy decision FE	Service authentication and authorization FE	Service authentication and authorization FE in NGN ensures that the end user has valid utilization rights for the requested service, and performs policy control at the service level.
7	Resource adaptor FE	User signalling interworking FE, Network signalling interworking FE	The two FEs in NGN have the responsibility for the interworking for different types of application signalling, and these also the functions of the resource adaptor FE in NGN-SIDE.

Table II.1 – Mapping between NGN-SIDE functional entities and NGN functional entities

No.	NGN-SIDE functional entity	NGN functional entity	Remarks
8	Access control FE	Application gateway FE	Both provide a secure open interfaces for the applications to use the resources.
9	Policy decision FE	Policy decision FE	Both make decision regarding network resources and admission control bases on network policy rules, SLAs, and service information.

Bibliography

[b-ITU-T I.312] ITU-T Recommendation I.312/Q.1201 (1992), *Principles of intelligent network architecture*.

[b-ITU-T Y.2001] ITU-T Recommendation Y.2001 (2004), *General overview of NGN*.

[b-ITU-T Y.2011] ITU-T Recommendation Y.2011 (2004), *General principles and general reference model for Next Generation Networks.*

[b-ITU-T Y.2201] ITU-T Recommendation Y.2201 (2009), *Requirements and capabilities for ITU-T NGN*.

[b-ITU-T Y.2261] ITU-T Recommendation Y.2261 (2006), *PSTN/ISDN evolution to NGN*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
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	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	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
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 and next-generation networks**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2013