3GPP TR 23.810 V8.0.0 (2008-09)

Technical Report

3rd Generation Partnership Project;
Technical Specification Group Services and System Aspects;
Study on Architecture Impacts of Service Brokering
(Release 8)

Keywords

Architecture, IMS, Services

3GPP

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet

http://www.3gpp.org

Contents

Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The objective is to study if there is enhancement needed to the current service interaction management architecture (e.g. SCIM as part of AS and Service Broker as part of OSA SCS) in order to satisfy requirements in TS 22.228 [1].

This technical report contains the results of a study on service interaction management architecture with focus on the following aspects:

- considering through scenarios, if the current service interaction management architecture sufficiently manages interactions between application servers, within the Home Network or on third party servers, and
- determine, through a functional element architecture study, whether there is a need for enhancement of the architecture, as well as any required extensions to the current IMS protocols and procedures.

In addition, alternative mechanisms to improve preventing interacting services may be identified.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2.

[2] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[3] 3GPP TS 23.002: "Network Architecture".

[4] 3GPP TS 29.328: "IP Multimedia (IM) Subsystem Sh Interface; Signaling Flows and Message Contents".

[5] 3GPP TS 29.228: "IP Multimedia (IM) Subsystem Cx and Dx Interfaces; Signaling Flows and Message Contents".

[6] 3GPP TS 23.218: "IP Multimedia (IM) session handling; IM call model; Stage 2".

3 Definitions and Abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [2] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [2].

IMS Service: An IMS service resides on an Application Server and can be invoked via initial Filter Criteria from S-CSCF.

IMS Service Interaction: When executing more than one IMS services in the same session, an IMS service interaction occurs when one or more of those services interact in a way that may differ from the expected behaviour when they are executed separately.

IMS Service Integration: Once interactions among multiple IMS services can be appropriately managed, it enables integration of multiple IMS services for better customer experience.

Service Broker: Logical function that manages service interactions among services hosted on a single or multiple Application Servers of any of the following types: IMS Application Servers (i.e. IM-SSF, SIP AS, OSA SCS).

Static IMS Service Interaction Management: IMS Service interaction management is static if the management of the interactions is based on the predictable and predefined service detection and resolution information.

Dynamic IMS Service Interaction Management: IMS Service interaction management is dynamic if the management of the interactions is based on dynamic information, e.g. the content of a message or the results from run-time.

Centralized Service Brokering Function: A centralized Service Brokering function manages service interactions and integration in a single entity of Service Broker.

Distributed Service Brokering Function: A distributed Service Brokering function constitutes multiple entities of Service Broker.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [2] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [2].

ISB	IMS Service Broker
SB	Service Broker

4 Architecture Requirements

4.1 General

The service brokering functions are to provide an end user a coherent and consistent IP multimedia service experience when multiple IP multimedia applications are invoked in a session. Such support involves identifying which applications are invoked per subscriber, understanding the appropriate order of the set of applications, and resolving application interactions during the session (TS 22.228 [1]). The applications can reside in any type of IMS Application Servers including an IM-SSF, SIP AS, OSA SCS or other (e.g. OMA enabler, Web Server) or any combination of the above.

Service brokering functions can be divided into two categories: on-line and off-line. Off-line functions include the following tasks:

1. Identify all applications subscribed by a user;
2. Understand how many ways these applications may work together by resolving their potential interactions;
3. Decide one or more service behaviours of combined applications (based on the user's expectation) for provisioning.

On-line functions then are to ensure that in a live session, when these multiple applications are invoked by the user, they will work as what the user expects them to work. This study covers the architecture impacts of the on-line service brokering functions, that is, how to provide architecture support to enforce the appropriate order of application execution with the guarantee of both security and charging.

On-line service brokering functions are used to resolve Static or Dynamic IMS Service Interactions. The S-CSCF iFC procedure is an on-line service brokering function for resolving static IMS service interactions whereas the Service Broker under this study item is an on-line service brokering function for resolving dynamic IMS service interactions.

The goal is to study the following potential requirements with an appropriate Service Broker architecture:

- The impacts of introducing the service brokering function to IMS core network and AS should be minimized

- The service brokering architecture should be flexible enough to process the potential interaction requirements for new applications.
- The service broker shall efficiently interact with the AS and avoid unnecessary interaction.
- Manage service interactions between IMS applications, enablers, and other non-IMS applications, potentially deployed over different application servers:
  - the architecture shall manage service interactions among all applications deployed over different types of application servers so that there will be no unexpected service behaviours.
- Support integration of services with existing IN services (e.g. CAMEL):
  - the architecture should accommodate both existing IN services and newly defined IMS services and support their integration.
- Allow service integration between SIP and non-SIP applications available via the IMS service architecture.
- Support service integration across multiple providers:
  - the architecture should support service integration over application servers of different providers.
- Allow users to personalize and control their services:
  - the architecture should allow end users to personalize and control how applications work together when there are multiple choices of integration available.
- The service broker should support service integrations across network hosted applications where the applications can reside either in the same AS or in different AS's.

Table 4.1 shows which of these general requirements are within scope of the current phase of the study.

Table 4.1 Service Broker Requirements – Phased Approach

SB General Requirements	Current Phase of the Study	To be addressed by future phases of the study
The impacts of introducing the service brokering function to IMS core network and AS should be minimized	In Scope	
The service brokering architecture should be flexible enough to process the potential interaction requirements for new applications. The SB should not route any unnecessary messages to an AS.	In Scope	
The service broker shall efficiently interact with the AS and avoid unnecessary interaction.	In Scope	
Manage service interactions between IMS applications, enablers, and other non-IMS applications, potentially deployed over different application servers – the architecture shall manage service interactions among all applications deployed over different types of application servers so that there will be no unexpected service behaviours.	Only service interactions between IMS applications are in scope. See NOTE.	Candidate extension
Support integration of services with existing IN services (e.g. CAMEL) – the architecture should accommodate both existing IN services and newly defined IMS services and support their integration.	Only service interactions between IMS applications are in the scope. See NOTE.	
Allow service integration between SIP and non-SIP applications available via the IMS service architecture.	Beyond Scope	Candidate extension
Support service integration across multiple service providers – the architecture should support service integration over application servers of different service providers (e.g. IMS operator).	Beyond Scope	Candidate extension
Allow users to personalize and control their services – the architecture should allow end users to personalize and control how applications work together when there are multiple choices of integration available.	Beyond Scope	
The service broker should support service integrations across network hosted applications where the applications can reside either in the same AS or in different AS's.	In Scope	
NOTE: With the focus only on IMS and SIP services, the only interface we have to be concerned about between the ISB and the AS is the ISC.		

4.2 Application Interaction Considerations

4.2.1 Application Interaction Scenarios

Application interactions can occur in one or more of the following scenarios:

1) Application interaction between application servers:

   Considering different application providing entities, the application interaction can be:

   - Interaction between networks hosted applications where the applications can be in the same AS or in different AS's.

2) Application interaction between providing entities:

   For IMS service brokering, applications provided by at least the following entities should be considered in the application interaction. They are:

   - IMS Applications provided by the SIP Application Server as defined in TS 23.218 [6], clause 5.1. These applications can be connected to the Service Broker.

   - OSA applications provided by the OSA Application Server as described in TS 23.218 [6], clause 5.1. These applications can be connect by an OSA API to the OSA Service Capability Server (SCS) and then connected to the Service Broker. The OSA SCS can control the invocation of the OSA applications.

- IN applications provided by the Service Control Point (SCP) can be connected by IN protocol such as CAP to the IN gateway (i.e. IM-SSF as described in TS 23.218 [6], clause 5.1) and then connected to the Service Broker, where the IN gateway can control the invocation of the IN applications.

3) Application interaction between application types.

   The applications can be of the following types:

   - Subscribed Applications. i.e. Applications to which the user subscribes and for which the user requires a subscription with a service provider;

   - Unsubscribed Applications. i.e. Applications that require no subscription.

   Considering application types, the application interaction includes:

   - Interaction between subscribed applications;

   - Interaction between subscribed application and unsubscribed application;

   - Interaction between unsubscribed applications.

4) Application interaction between Application Status:

   Applications can have the following status:

   - Invoked;

   - Not invoked.

   Considering the relationship between application interaction and the application status, the application interaction includes:

   - Interaction between an application that has already been invoked and an application that will be invoked;

   - Interaction between applications that have already been invoked.

5) Application interaction between involved parties:

   The applications can be provided to the same user or different users, so the application interaction includes:

   - Interaction between applications for the same user;

   - Interaction between applications for different users.

NOTE: Interaction between applications for different users is out of scope.

6) Application interaction between involved sessions:

   In the communication, the user can be engaged in multiple-sessions, in this case the application interaction would include:

   - Interaction between applications invoked in one session;

   - Interaction between applications invoked in multiple sessions.

7) Application interaction between service provider domains:

   The applications or application server can belong to different service providers, in which case the application interaction includes:

   - Interactions between applications provided by different service providers;

   - Interactions between applications provided by the same service provider.

Table 4.2.1 shows which of these application interaction scenarios are within scope of the current phase of the study.

Table 4.2.1 Application Interaction Scenarios – Phased Approach

Scenarios of Service Interaction	Current Phase of the Study	To be addressed by future phases of the study
Application interaction between application servers	In scope	
Application interaction between providing entities	In scope. See NOTE	
Application interaction between application types	Only Interactions between subscribed applications in scope	Interaction between unsubscribed application is a a candidate extension
Application interaction between application status	Both Invoked and not Invoked application interactions are in scope	
Application interaction between involved parties	Only Interactions between applications for the same user are in scope	Interactions between different users is a candidate extension
Application interaction between involved sessions	Only Interactions between applications invoked in one session are in scope	Interactions between applications invoked in different sessions are a candidate extension
Application interaction between service provider domains	Only Interactions between applications provided by the same service provider are in scope.	Interactions between applications provided by different service provider (e.g. IMS operator) are a candidate extension
NOTE: The ISB focuses only on IMS and SIP services in the current phase of the study and covers only three types of IMS application servers: IM-SSF, SIP AS, OSA SCS Application Server.		

4.2.2 Application Interaction Types

The application interaction type can be divided into two classes: undesired application interaction and applications coexist:

1) Undesired application interaction:

   This type of application interaction can cause an undesired result, and should be avoided. It includes:

   - Applications conflict: application features can not be realized when they work at the same time.
   - Undesired application co-work: applications may be able to work together, but it is recommended to inhibit one application feature.
   - Undesired application inhibit: the process of one application will cause the inhibition of another application, which is not desired.

2) Application coexist

   Applications can be work together and will not invoke any negative effect. Normally it includes these aspects:

   - Interaction with a priority requirement: This includes the priority of the applications in the same AS and the priority of the applications across different AS.
   - Interaction with chaining requirement:
     1) Fixed chaining: the application invocation order is fixed in each communication session.
     2) Dynamic chaining: the application invocation order can be changed dynamically in different communication sessions based on the dynamic information other than normal SPT.

Table 4.2.2 shows which of these application interaction scenarios are within scope of the current phase of the study.

Table 4.2.2 Application Scenarios - Scope

Service Interaction Types	Current Phase of the Study	To be addressed by future phases of the study
Service interactions without resolution	Out of scope of SB	
Service interactions resolved by priority (precedence order)	In scope	
Service interactions resolved with fixed chaining	In scope	
Service Interactions resolved with dynamic chaining	In scope with focus on single session, single user, and single provider services. See NOTE.	Candidate extension of dynamic chaining across multiple sessions, multiple users, and multiple service providers
NOTE: The current phase of the study ISB offers the service interaction management capabilities beyond static precedence order and fixed chaining offered by the existing initial Filter Criteria (iFC) mechanism in S-CSCF. ISB is a first step to offer dynamic precedence order and dynamic chaining of IMS SIP services in the IMS architecture.		

4.3 Security

The solutions proposed for Service Broker should take into consideration the following requirements:

- When Application Servers reside in different domains or are hosted by different service providers, the solution should provide adequate security mechanisms if these application servers reside outside the trusted domain.
- When Application Servers and the network entities performing the Service Brokering function reside in different domains or are hosted by different service providers, the solution should provide adequate security mechanisms if Application Servers and the network entity/entities performing the Service Brokering function reside outside the trusted domain.
- Existing 3GPP security should be taken into account.

In this phase of the study applications interactions between different providing domains are out of scope, therefore security considerations when Applications Servers reside in different domains or when the Service Brokering function and Application Servers are in different domains are out of scope of this phase of the study.

4.4 Charging

The solution must consider charging implications of the architecture and its potential to support the coordination of charging events from multiple applications deployed on multiple Application Servers. Both online and offline charging scenarios must be considered.

When resolving application interactions, the SB functions may alter the sequence and or invocation/or not of certain applications. The sequence in which applications are invoked shall be reflected in the charging information.

5 Architecture Alternatives

5.1 Service Interaction Management by Service Brokers

The Service Brokering Functions under consideration can be centralized on a single Service Broker, distributed, or hybrid (i.e. both centralized and distributed) in order to manage the interactions among multiple Application Servers.

5.1.1 Architecture Alternatives to Interaction Management by Service Brokers

Two service interaction management scenarios are considered:

- Centralized service interaction management:

- where a centralized Service Broker is used to coordinate and control the interactions among multiple interacting applications.

- Distributed service interaction management:
  - where the Service Brokers with service brokering functions coordinate and control the interactions among multiple interacting applications.

In addition, the mixed use of centralized and distributed service interaction management to support a hybrid architecture is also considered.

5.1.1.1 Centralized Service Brokering Functions

In this architecture, the Application Servers involved in offering the integrated service are unaware of the existence of the Service Broker and the S-CSCF views the Service Broker as an Application Server supporting the ISC interface. The Service Broker Functions can be located outside S-CSCF, or embedded in S-SCCF.

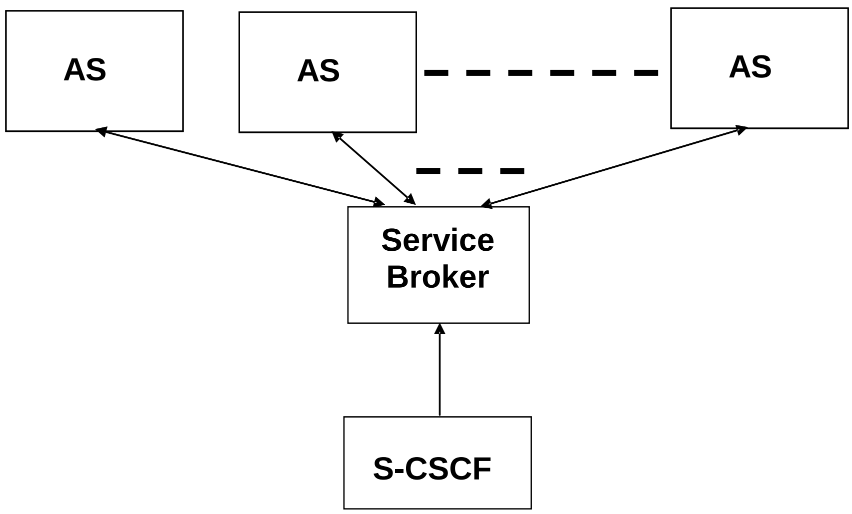

Figure 5.1.1.1: Centralized Service Broker

The interfaces between the Service Broker and the Application Servers continue to be ISC.

Standards thus need to be defined for the Service Broker including its interfaces and procedures.

5.1.1.2 Distributed Service Brokering Functions

In this architecture, each Application Server involved in offering the integrated service is equipped by one Service Broker, which may be located independently or embedded in the AS, so that they can coordinate to handle the services involved. The S-CSCF views each Service Broker as one Application Server supporting the ISC interface. The S-CSCF relays the messages among the Service Brokers until all Application Servers finish their functions.

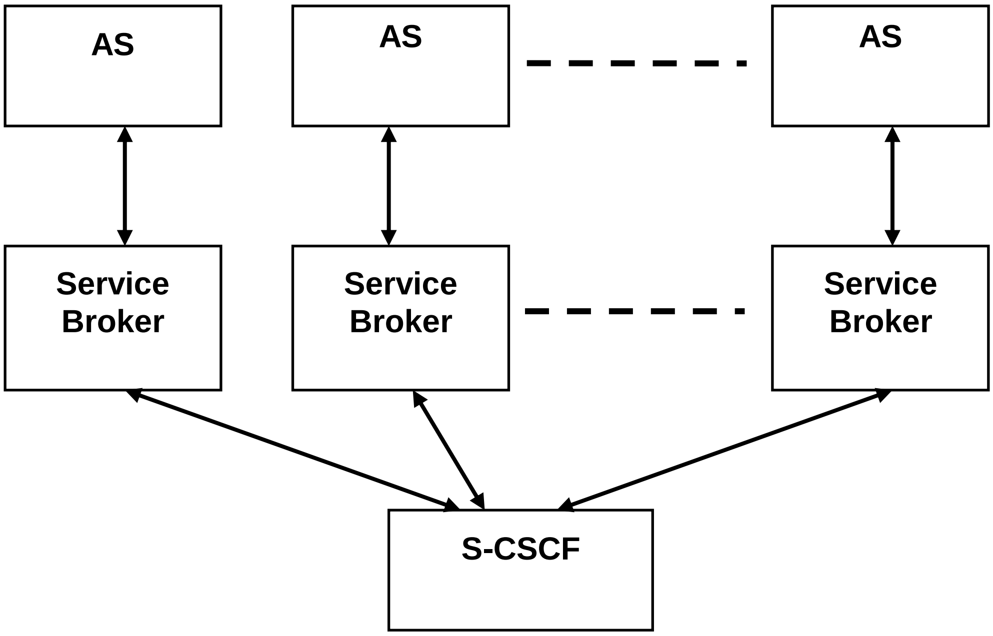

Figure 5.1.1.2: Distributed Service Broker

In order to interwork multiple Service Brokers consistently and coherently, standards are required for protocols and procedures of these distributed brokering functions.

5.1.1.3 Hybrid Service Brokering Functions

This architecture is a hybrid of the above two architectures. The Service Brokers under this architecture have to manage service interactions not only among the application servers under its direct control but also with its peer Service Brokers.

Two possible configurations of the hybrid architecture are depicted below. Note that these are not supposed to be exhaustive as there are many possibilities of hybrid configuration. These two are just examples of many possibilities.

1) Architecture Configuration 1 where some server brokers (e.g. the rightmost one) act as both centralized and distributed service brokers.

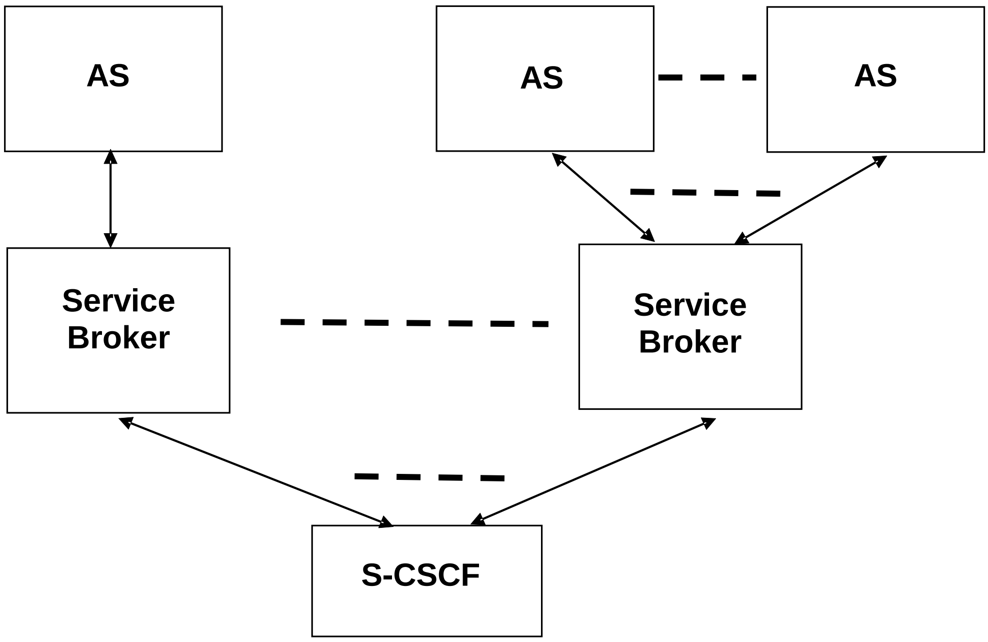

Figure 5.1.1.3-1: Hybrid Service Broker (1)

2) Architecture Configuration 2 where multiple service brokers are interfaced with the S-CSCF and they act as both centralized and distributed brokers.

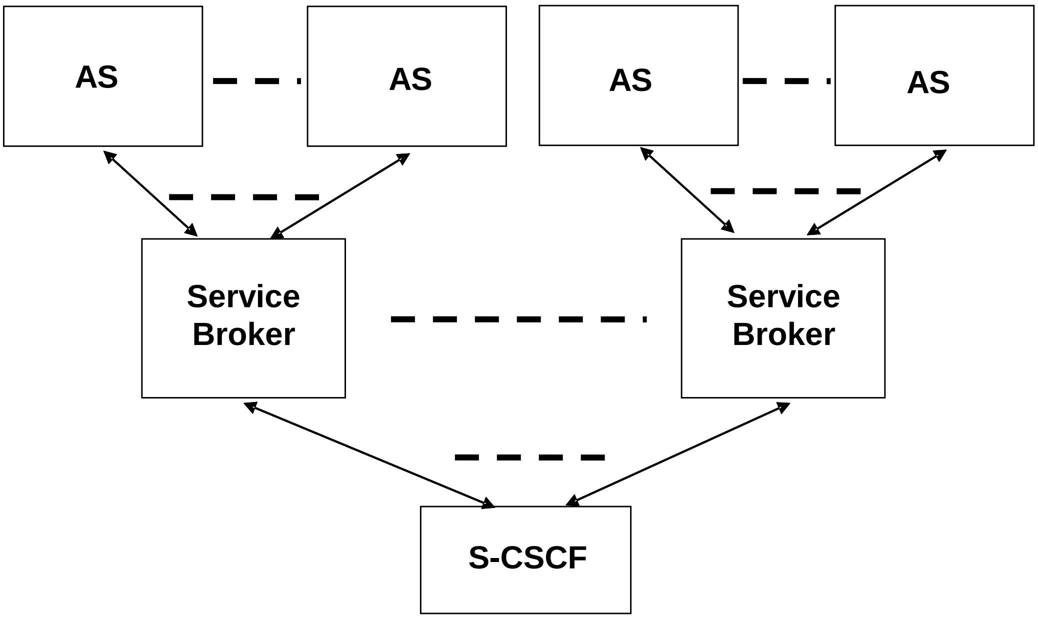

Figure 5.1.1.3.2: Hybrid Service Broker (2)

5.1.1.4 Architecture Alternatives Evaluation

The architectural alternatives in the previous clauses are all for single component service interactions (i.e. limited to an S-CSCF under a single user, single session, and single provider domain environment). Nevertheless, to further simplify the problem scope we conclude that initially there are no distributed service brokering functions but a centralized service brokering function to manage service interactions and integration (i.e. figure 5.1.1.1 of clause 5.1.1.1). The other architectural alternatives including both distributed service brokering functions and hybrid service brokering functions will be FFS.

5.1.2 Interaction Logic

The Service Broker manages the execution of the services or applications based on service Interaction Logic. Interaction logic may be captured by a set of rules based on the service invocation history that is a list of (Service ID, Service Effect). The SB manages the execution of the services or applications based on these rules. The interaction logic for Service Broker can be provisioned on the HSS as part of the subscriber's profile.

The following use case demonstrates how iFCs in the S-CSCF and interaction logic in the Service Broker work together to manage IMS service interactions and integration.

Assume via off-line service interaction detection, the following dynamic chaining of services is developed for the subscriber:

Depending on the result of Service A on AS1, if Service A returns success, do Service C on AS3 followed by Service D on AS 4 and stop SIP routing. Otherwise, do Service B on AS2 and if Service B returns success, also do Service C on AS3 followed by Service D on AS 4 and stop SIP routing; otherwise, do nothing but continue SIP routing.

So there are three ways of chaining the services:

(1) Service A (AS1) – Service C (AS 3) – Service D (AS4) – stop routing,

(2) Service A (AS1) – Service B(AS2) - Service C (AS3) – Service D (AS4) – stop routing, or

(3) Service A (AS1) – Service B(AS2) – continue routing,.

To make this happen, we provision the iFC based on the following priority order:

- Service A – Service B – Service C – Service D

Then we provision the following interaction rules in the Service Broker:

- IF History contains (Service A, Success), SKIP Service B
- IF History contains (Service B, Failure), SKIP Service C
- IF History contains (Service B, Failure), SKIP Service D

5.1.3 Equivalent Classes

Services or applications residing on the AS may be classified into equivalent classes based on their impact on other services or applications. Each class is then assigned a unique Service (Class) ID. Interaction logic may be captured by a set of rules based on the service invocation history that is a list of (Service ID, Service Effect). The SB manages the execution of the services or applications based on these rules.

Standardization of equivalent classes for services/applications is beyond the scope of the current phase of the study. Each operator can define its equivalent classes.

5.1.4 Architecture Reference Model

An architecture reference model of the Service Broker is depicted below. In this reference architecture, the Service Broker (SB) has the following interfaces:

- ISC interface to the AS (e.g. SIP AS, IMS-SSF, OSA-SCS) and the S-CSCF. This interface needs to be enhanced to carry the history of service invocations (i.e. Service ID and Service Effect);
- Sh or Cx interface to the HSS to download the interaction logic.

This architecture can be applied to various types of IMS service brokering functions regardless whether they are centralized, distributed, or hybrid:

- For centralized service brokering functions as depicted in clause 5.1.1.1, the S-CSCF will interface with only one SB, the SB interfaces with multiple ASs.
- For distributed service brokering functions as depicted in clause 5.1.1.2, the S-CSCF interfaces with multiple SB and each SB interfaces with only one AS.
- For hybrid service brokering functions as depicted in clause 5.1.1.3, each S-CSCF may interface with multiple SBs while each SB may also interface with multiple ASs.

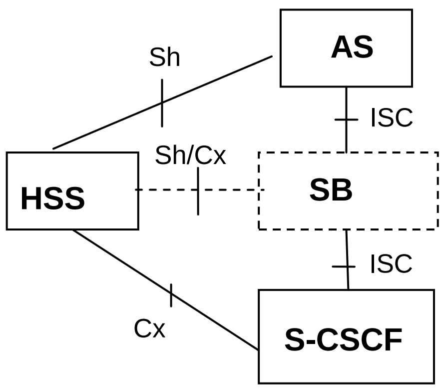

Figure 5.1.3-1: Reference Architecture

The SB is represented as a dotted box to indicate that it can be either a standalone network entity, i.e. a separate component from the AS and the S-CSCF, or it can be implemented as part of the AS or the S-CSCF. The interface between the SB and the HSS is also represented as a dotted line to indicate that this is an optional interface as the interaction logic for the SB can be downloaded from the HSS or can be provisioned locally in the SB.

As the current phase of the study is scoped to deal with centralized service brokering functions only, the reference architectural in figure 5.1.3-1 can be expanded into the functional architecture of a centralized Service Broker as follows.

- In general, one SB will manage multiple Application Servers.
- The SB can be a standalone IMS component or be part of the S-CSCF as indicated by dotted box.
- The interface between the HSS and the SB is optional as indicated by dotted line.

NOTE: When the SB is part of the S-CSCF, the Cx interface will be used. When the SB is part of the AS, the Sh interface will be used.

Editor's note: When the SB is a standalone component, whether to use the Sh or the Cx interface is FFS.

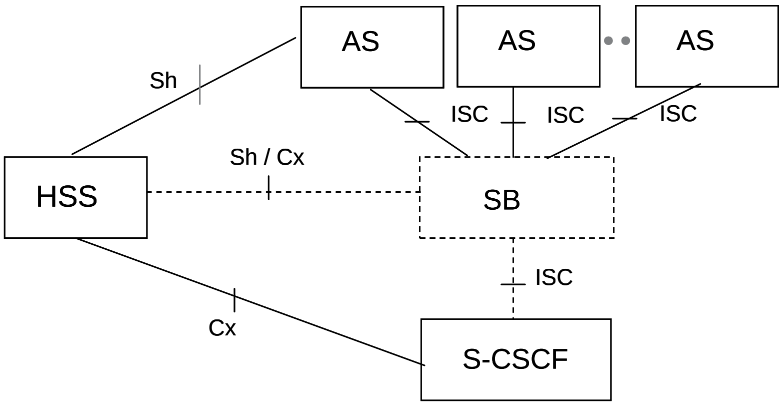

Figure 5.1.3-2: Current phase of the study Service Broker Architecture (Centralized SB for a single S-CSCF)

5.1.5 Identified Impacts to Current Architecture and Interfaces

The impact of the SB reference architecture on the current IMS architecture (see TS 23.228 [3]) and interfaces is as follows:

The SB is added as a new functional component in the IMS architecture and can be a separate component from the AS and the S-CSCF, or it can be implemented as part of the AS or the S-CSCF. The SB has the following interfaces to other IMS entities:

- ISC interface between the SB and the AS (e.g. SIP AS, IMS-SSF, OSA-SCS) and between the SB and the S-CSCF. Both ISC interfaces will be enhanced to carry IMS service invocation history that is expressed as a list of (Service ID, Service Effect).
- The Sh interface (see TS 29.328 [4]) or Cx interface (see TS 29.228 [5])between the SB and the HSS to download the interaction logic. However, this interface is optional and it is not needed when interaction logic is provisioned on the SB.

Editor's Note: Harmonization with ISC enhancements in clause 5.5.2 is FFS.

When the interaction logic is not provisioned on the SB and the SB downloads it from the HSS through the Sh (or Cx) interface, the Sh (or Cx) interface should carry the following information:

- Interaction Logic expressed as a set of rules based on service invocation history.

The role and use of iFC may need to be aligned with the notion of equivalent classes of services or applications. This is FFS.

NOTE 1: When the SB is part of the S-CSCF, the ISC interface between the SB and the S-CSCF is internal to the S-CSCF.

NOTE 2: When the SB is part of the AS, the ISC interface between the AS and the SB is internal to the AS.

5.1.6 Architecture Impacts Evaluation

In summary, the Service Broker adds additional capabilities to static service brokering functions provided by the initial Filter Criteria in the S-CSCF and enables the possibility of dynamic service brokering functions in IMS. Given a set of IMS applications, depending on the subscriber's requirements there may be multiple sequences or ways of integrating these applications. With the initial Filter Criteria, only a static order of chaining these applications is possible. The Service Broker, via the service invocation history in the ISC and rule-based interaction logic provisioned offline, can provide this additional function of dynamically chaining the IMS services at the run time based on service invocation history. The existing iFC mechanism in S-CSCF will continue to be utilized and play a key role in enabling these dynamic service brokering functions.

Use Case: Given two services Service A and Service B on AS1, one service, Service C on AS2, and one service, Service D on AS3. Suppose we need to support dynamic chaining of these services in the following way:

- Service A on AS1 – Service C on AS2 – Continue.
- Service A on AS1 – Service D on AS3 – Service B on AS 1 - Stop.
- Service A on AS1 – Service B on AS1 – Stop.

Which of the above will be executed depends on the result of Service A that can be a, b, or c.

To provision the above services, we define the initial Filter Criteria in the S-CSCF and the interaction logic in the SB, respectively, as follows:

- iFC: Service A on AS1 – Service C on AS2 – Service D on AS3 – Service B on AS1.
- Interaction Logic:

  Rule 1 – IF History contains (Service A, Result a), SKIP Service D and Service B.

  Rule 2 – IF History contains (Service A, Result b), SKIP Service C.

  Rule 3 – IF History contains (Service A, Result c), SKIP Service C and Service D.

5.1.7 Other Enhancements

When the interaction logic is stored on the HSS and downloaded, the user profile could be used to store this information.

5.1.8 Security

In this phase of the study applications interactions between different providing domains are out of scope, there are no security considerations to take into account.

5.1.9 Charging

Editor's note: This clause presents how <solution architecture 1> addresses the charging requirements in clause 4.2.

5.2 Use Case to Illustrate Functions of ISB

Let's consider a user profile containing 4 initial filter criteria for triggering:

- A denied termination (DT) presence service.
- An incoming call barring (ICB) service.
- An incoming call logging (ICL) service.
- An e-mail notification (EN) service.

Each of the above services is provided by a separate AS.

The service desired is as follows:

- If the user sets the presence to be "do not disturb", the IMS will bar all the incoming requests, but record the information of requester (i.e. requester's Caller ID) and send an e-mail notification to alert the user. That is, the chaining of the services is DT-ICL-EN-stop.
- If the user turns off "do not disturb" in the presence, all the incoming requests will be screened against a black list by the IMS. If the requester is in the black list, the incoming request will be barred while the information of requester (i.e. requester's Caller ID) is logged and an e-mail notification is sent to alert the user. That is, the chaining of the services is DT-ICB-ICL-EN-stop.
- If the incoming request is not barred at all, no need to log the incoming request or to send an e-mail notification to the user. The request will be routed normally to the user (either via INVITE or MESSAGE). That is, the chaining of the services is DT-ICB-continue.

We thus provision four iFCs to support the above service with the following order:

a) iFC1 – AS1 hosting the denied termination (do not disturb) presence service.

b) iFC2 – AS2 hosting the incoming call barring service.

c) iFC3 – AS3 hosting the incoming call logging service.

d) iFC4 – AS4 hosting the e-mail notification service.

The current IMS Initial Filter Criteria cannot support such dynamic chaining of four services DT, ICB, ICL, and EN. The dynamic chaining occurs because the execution order of the services can be:

(1) DT-ICL-EN-stop,

(2) DT-ICB-ICL-EN-stop, or

(3) DT-ICB-continue.

This service interaction can only be resolved by the Service Broker enhanced with history-based dynamic chaining capability.

a) ISC is enhanced to carry the service invocation history. Information carried includes what service has been invoked and the result of invocation (e.g. the call has been denied or screened) if needed.

b) Rules can be defined in Service Broker for managing service interactions based on this history information. For example, in the use case above we just need two simple rules:

  - If history includes "DT – on", skip "ICB".
  - If history includes "ICB – call not barred", skip either "ICL" or "EN".

NOTE: For the above service to work, when a call is barred, the ISC enhancement identified in clause 5.5.3 is required for the S-CSCF to continue to evaluate the remaining iFCs.

ISC enhancements such as those described in clause 5.5.2 may be needed to support this scenario.

5.3 ISC improvements

The IMS service interaction management architecture may be able to be improved through enhancements to the IMS Service Control interface and supporting architecture (iFCs).

5.3.1 Improvement when Retargeting R-URIs

5.3.1.1 Problem Description

During terminating call handling, the R-URI identifies both the served user/UE (the user/UE that the S-CSCF is serving) and the target user/UE (the user/UE that the session is finally destined towards). As such, if a SIP-AS that is performing terminating services retargets the R-URI changes the R-URI in any manner then information about the served user/UE is lost and subsequent filter analysis in the S-CSCF is terminated and the S-CSCF forwards the SIP request towards the new target. This has the effect of not linking in other application servers that may have been interested in the SIP request. This includes the case where a terminating SIP-AS changes an IMPU to a GRUU.

This is illustrated below.

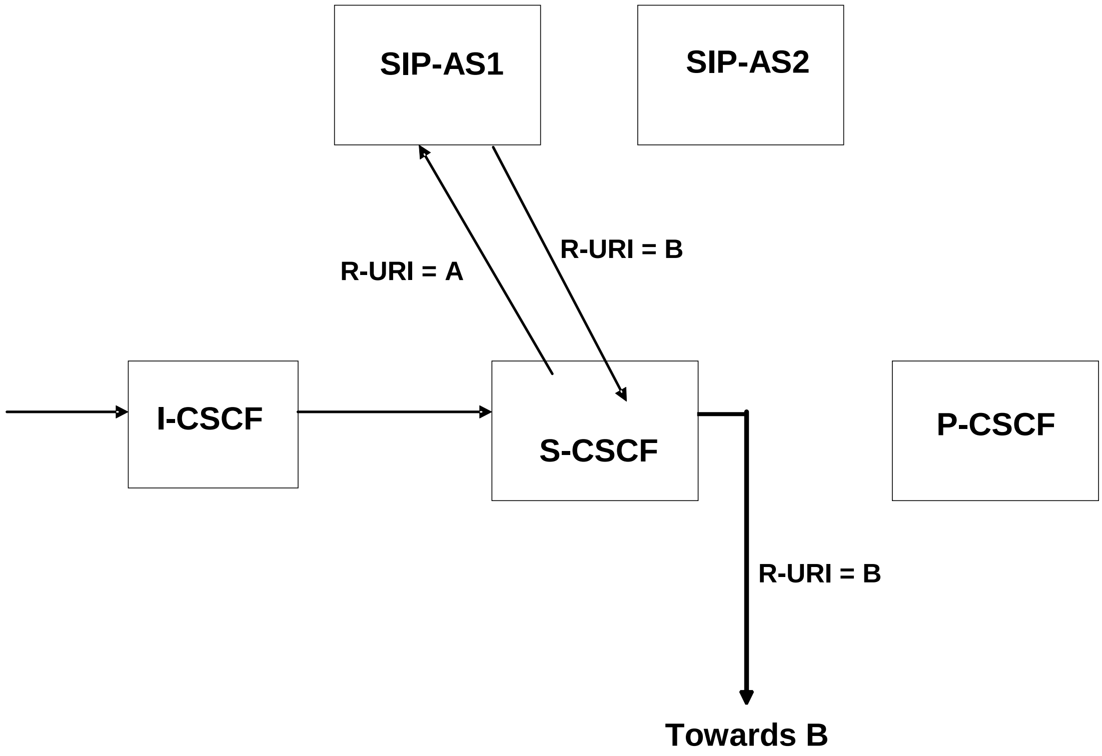

Figure 5.5.1.1: Example of Retargeting the R-URI

In the figure above, for a call that would normally be terminated towards the user/UE identified through the R-URI of "A", the S-CSCF would route the call through both SIP AS1, SIP-AS2 before forwarding towards the user. If, however, SIP-AS1 changes the R-URI, the S-CSCF will not in-link SIP-AS2, but instead forward route the request towards the user/UE identified as R-URI=B. This includes the case where SIP-AS1 updates the request URI from an IMPU to a GRUU.

It should be noted that if SIP-AS1 does re-target the R-URI, then SIP-AS 2 needs information about both the target User/UE and the user/UE who the S-CSCF is serving. This also requires changes to the SIP-AS2 invocation after the R-URI is modified.

5.3.1.2 Potential Solution: Separation of the served and the target UE information

One solution to this problem identified could be to separate the information regarding the served user from the target user/UE information over the ISC. The means to transport and the format of such a request is a stage 3 issue.

Editor's note: Interaction with other SIP-AS need to be studied, e.g., communication Diversion services and communication call bar services.

5.3.2 Improvement for Incompatible Services

5.3.2.1 Problem description

Let's consider a user profile containing 3 initial filter criteria for triggering:

- A freephone service.
- A voice-activated dialling service.
- An outgoing call barring service.

When making an outgoing call, the user decides to speak or dial, on per-call basis. If the user dials a service access code to the voice-activated dialling service, the first initial filter criteria will be evaluated but will not match. The second initial filter criteria will be evaluated and will match. The AS hosting the voice-activated dialling service will ask the calling user to speak the name of the person he wants to call and translate this name into a destination number. According to the current IMS procedures, the third trigger will be evaluated when the AS returns an INVITE request with the destination number. The third trigger will match and the outgoing call barring service may reject the call if the user is not allowed to place outgoing calls to this number.

This is an acceptable behaviour.

If the user dials a freephone number, the first initial filter criteria matches and the INVITE request shall be routed to the appropriate AS. The AS translates the freephone number into a geographical number that can be used to route call to the appropriate location. According to the current IMS procedures, the second and third triggers will be evaluated when the AS generates an INVITE request with the translated number. The third trigger will match and the outgoing call barring service may reject the call if the user is not allowed to place outgoing calls to the area in which the actual destination is located.

This is not an acceptable behaviour.

5.3.2.2 Potential Solution: Addition of Compatibility Class to Initial Filer Criteria

To allow the S-CSCF to handle simple services interaction, such as avoiding to trigger the service corresponding to two incompatible services during the same sessions, the notion of class of compatibility could be introduced. This class of compatibility would be contained in the iFC information stored in the HSS and downloaded to the S-CSCF, and would indicate to the S-CSCF which iFC should not be triggered after other iFCs has been successful invoked , and the S-CSCF should obtain the actual service invocation status information.

The S-CSCF decides if a service was Successfully invoked based on two criteria:

1) whether the actually iFC was triggered or not, and
2) whether the AS in case the iFC was triggered, returns an error or not or no response at all.

i.e., Successfully invoked is when an AS is triggered and does not return any error response. In the case the AS returns an error or no response at all, it will be seen as an unsuccessful invocation and the procedures of clause 5.5.3 may be applied to decide whether to continue or not.

NOTE 1: For terminating services, if the Request-URI changes during processing, the information about successful invocation may be lost.

For example, in the figure below, the user profile contains 4 iFCs. Each of those iFC has been assigned a compatibility class:

- iFC1 has been assigned a compatibility class of 1.
- iFC2 has been assigned a compatibility class of 1.
- iFC3 has been assigned a compatibility class of 2.
- iFC4 has been assigned a compatibility class of 3.

The number of set of compatibility class needed is defined by the operator. The rules of allowed interaction between those different classes are pre-configured in the S-CSCF by the operator. Those set of rules could be defined for example as follows:

- COMPATIBILITY_RULE = COMPATIBILITY_CLASS, LIST_OF_NON_COMPATIBLE_CLASSES
- LIST_OF_NON_COMPATIBLE_CLASSES = *{COMPATIBILITY_CLASS}

and stored in the S-CSCF (e.g. in an XML file) and could be provisioned in the S-CSCF (e.g. by O&M mechanisms).

NOTE 2: The detailed structure of those set of rules is left for stage 3.

An iFC is validated if none of the previously triggered iFCs compatibility class are part of the list of non compatible classes. This means that if iFCa has been triggered then this will result in a subsequent iFC (e.g. iFCb) being disabled if iFCa is in a compatibility class that is considered non compatible with iFCb.

In our case, iFCs belonging to compatibility class 2 must not be triggered if services corresponding to iFCs belonging to compatibility class of 1 have been successful invoked before. At the same time, iFCs belonging to compatibility class of 3 can be triggered if no services corresponding to iFC belonging to set 2 has been successful invoked.

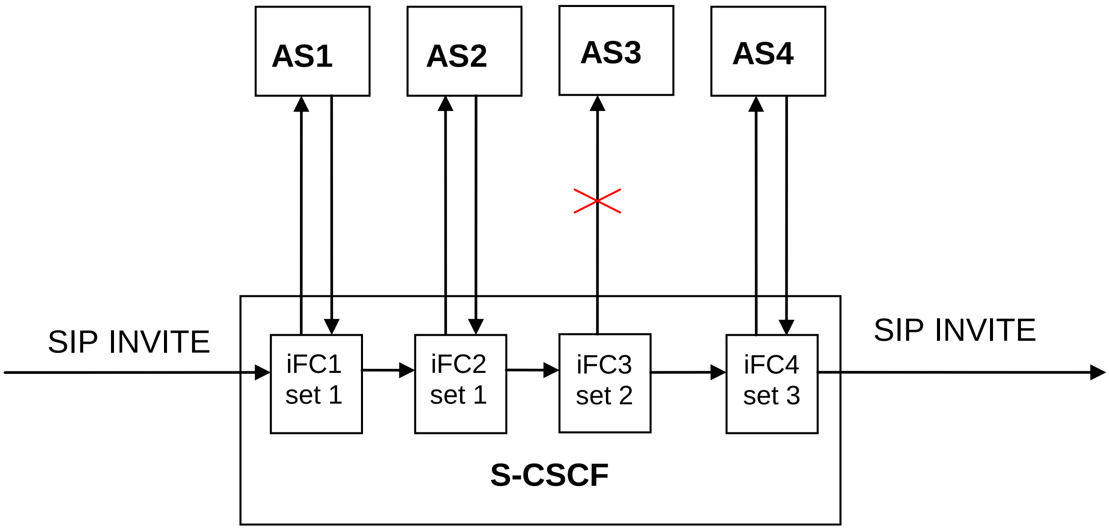

Figure 5.5.2.2-1: Example of iFC Compatibility Class Checking

The following aspects will need to be studied further: how to indicate the actual service invocation status.

5.3.3 The Handling of AS Generated Error Responses

5.3.3.1 Problem Description

During originating or terminating a session, if the S-CSCF receives any final response from the AS, it will forward the response towards the originating UE, without verifying the matching of filter criteria of lower priority and without proceeding for further steps.

However, under certain circumstances it may be desirable to invoke the succeeding AS if the service it hosts is immune to the fault and still meaningful to execute. The currently defined ISC/iFC behaviour lacks the means to handle this situation, as it is only possible to invoke an additional action when it is not possible to reach an AS.

5.3.3.2 Potential Solution: Enhancement of iFC

One solution is to update iFC to be able to express whether the session should be terminated or continued depending on the error response received. If the session is continued, the original request sent to the AS is used for the subsequent handling.

5.3.4 Enhanced Service Triggering Conditions

5.3.4.1 Problem Description

Under the existing specification SPT (service point trigger) only define five possible service triggering points, including Request-URI, SIP Method, SIP Header, Session Case and Session Description, which are all directly derived from the SIP initial request. At certain scenarios the services need route SIP request besides the existing SPT.

For example, User-A do an IMS registration using a handset supporting CSI capability. When the IMS network receives a terminating IMS service, the S-CSCF should invoke the CSI AS based on UE capability.

As the CSI is not a subscription service, if the CSI service has been implemented on the network and DSAI feature is not supported by the CSI AS then the terminated request should always route to the CSI AS no matter whether UE support CSI capability. But indeed the S-CSCF can get UE capability information from IMS registration procedure. If that information can be used by S-CSCF on the following SIP terminating process, it may be more efficient comparing to always route the request to the CSI AS.

Similar scenario can also be happened on the VCC case. It should be regarded as an IMS general issue.

5.3.4.2 Potential Solution

The new SPT (service point trigger) shall include the UE capability information. S-CSCF can get that information from IMS registration procedure and using that information for subsequent originating and terminating request processing.

When the S-CSCF on terminating side executes the UE capability SPT, the following procedures should be adopted to avoid interaction problems with forking:

1) If the Request-URI can be resolved to one contact address then the terminal's capability associated with that contact address should be used for iFC processing.

2) Otherwise the UE capability SPT should not be used for service triggering to avoid the selected AS is not match with the destination terminal's capability.

5.3.5 Indicating Specific Services Executed by the AS with Multiple Services Supplied

5.3.5.1 Problem Description

Normally multiple applications can be supplied in one AS. In order to identify correct application to be executed, the Application server class in iFC may contain specific identification of the application, such as "application @ as.home.net".

Although AS can supply multiple application with once invocation, and handle the priority sequence by itself, but when there have the priority sequence between part services of this AS and other As, it may be uncertain which application shall be executed next during once invocation of the AS. So normally AS will return the request to S-CSCF, let S-CSCF give a new indication of the application need be executed next based on iFC.

Considering the following the user case:

> The user A has the subscriptions to the AS1 with the outgoing call barring service (OCB), the customized coloured Image service, and the originating identification restriction service (OIR), and the AS2 with the free phone service, and the user A hope that the services executed order is the outgoing call barring service, the free phone service, the customized coloured Image service and the originating identification restriction service.
>
> The user B has also the subscriptions to the four services, but the user B hope that the services executed order is the OCB service, the OIR service, and the free phone service, the customized coloured Image service.
>
> It is clear that AS1 should not execute the other service automatically after the OCB service has been executed. S-CSCF need give AS an indication which service it should executed. Also special service logic identification should not be introduced. Otherwise every time the application order has been changed, the new special logic indication will be introduced. This will cause the large management between HSS and AS.

One way the user A and the user B both have the four iFCs, each iFC express one service to be executed, but this will add the unnecessary delay on the session setup time, for some application are executed continuously on the same Application server, but now it need return to S-CSCF then back.

5.3.5.2 Potential Solution

One solution is to update iFC to support that multiple service can be invoked by a single SIP transaction, such as update Application server Class of iFC to reflect multiple service need be executed. An element will be added to the Application Server Class to identify the services and the services execution order on an Application Server within a single SIP transaction.

5.3.6 Improvements to Support Service Broker Architecture described in clause 5.1

The architecture proposed in clause 5.1 requires enhancements to the ISC interface to carry history of invoked services and the result (or effect) of invoking an application. This information is needed by the Service Broker function to apply application interaction rules captured in the Interaction Logic.

6 Conclusion

6.1 ISC Improvements

6.1.1 Improvement when Retargeting R-URIs

It is recommended to separate the information regarding the served user from the targeted user/UE information over the ISC. The means to transport the information and the format of such a request is a stage 3 issue.

6.1.2 Improvement when Handling Application Server Generated Error Responses

It is recommended that iFC be enhanced to allow the marking of whether an error response from an applications server should result in the termination of the session and forwarding of the error onto the UE or whether the error should be discarded and processing continue to the next iFC in the list, with the original request sent to the Application Server used for subsequent handling.

6.1.3 Enhanced Service Triggering Conditions

It is recommended to add UE capability as one of SPT. S-CSCF can get that information from IMS registration procedure. When S-CSCF on terminating side executes the UE capability SPT, if the Request-URI can be resolved to one contact address then this type of SPT can be used for service triggering, otherwise this type of SPT should not be used.

6.1.4 Indicating Specific Services Executed by the AS with Multiple Services Supplied

It is recommended that ISC interface and the initial-filter criteria transported over the Cx interface allow support of the invocation of multiple services within a single SIP transaction by an Application Server. A possible implementation of this requirement over the Cx interface is to add an element to the Application Server Class of the iFC to identify the services and the service execution order.

6.2 Study Conclusions

The study has identified areas in which service interaction management could enhance the operation of networks. The study has resulted in a number of improvements to the current architecture in the existing specifications. It has also identified a number of architecture alternatives and further enhancements that provide a starting point for further development of solutions.

The study concludes that continuing the work with a focused scope to address the outstanding issues (e.g. architecture requirements not satisfied, other interface enhancements) related to service interaction management is recommended.

The study has concluded on proposal of a first phase and a second phase as described in tables in clause 4 and proposes to use that as a basis for further discussion.

Annex A: Change history

Change history								
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**Old**	**New**
2008-06	SP-40	SP-080354	-	-	-	MCC Editorial update for presentation to TSG SA for Information	0.10.0	1.0.0
2008-09	SP-41	SP-080548	-	-	-	MCC Update for presentation to TSG SA for approval.	1.0.1	2.0.0
2008-09	SP-41	-	-	-	-	MCC Update of approved TR for publication	2.0.0	8.0.0