FS – Provision Manager

Content

1 INTRODUCTION 2

1.1 GLOSSARY 2
1.1.1 Subscription 2

2 FUNCTION REQUIREMENTS (COMMERCIAL) 2

3 FUNCTION SPECIFICATION (DESIGN RELATED) 2

3.1 INTRODUCTION 2
3.2 EXPORTED INTERFACES 3
3.2.1 IProvisioning 3
3.3 IMPORTED INTERFACES 4
3.4 FUNCTIONS 4
3.4.1 CAI communication 4
3.4.2 CAI server Host 5
3.4.3 Subscription attributes 5
3.4.4 Creation of a subscriber, first request 5
3.4.5 Create using an established connection 6
3.4.6 Asynchronous/Synchronous mode 7
3.4.7 Fault scenarios 8
3.4.8 Events 10

4 EXTERNAL OPERATION CONDITIONS 10

4.1 CONFIGURATION 10
4.2 LOGGING 10

5 CAPABILITIES 10

6 REFERENCES 10

7 TERMINOLOGY 11

History

Version	Date	Adjustments
A	2006-10-03	First version. (MANDE)
B	2006-11-08	Removed req. of usage of localhost as MAILHOST in section 3.4.3. (MANDE)

1 Introduction

This document specifies the function of the Provision Manager component. The Provision Manager (PM) provides an interface that can be used by other components in order to create or delete a subscriber in the user directory in MUR. The creation of a subscriber allows the client to provide a number of attributes for the subscriber among which some are mandatory.

The Provision Manager communicates with MUR/MUP (?) using CAI in order to manage subscribers.

1.1 Glossary

1.1.1 Subscription

A subset of a subscribers profile that is possible to specify when adding a subscriber using the CAI interface (see ref. [1]).

2 Function Requirements (Commercial)

The following requirements have been identified:

- Automatic mailbox provisioning: It shall be possible for the system to automatically provision a new subscriber.

3 Function Specification (Design Related)

3.1 Introduction

The Provision Manager exports the IProvisioning interface. This interface is used by clients to create or delete subscribers in the subscriber database.

The Provision Manager imports the ILog, IConfig, ILocateService, IServiceInstance and IEventDispatcher interfaces.

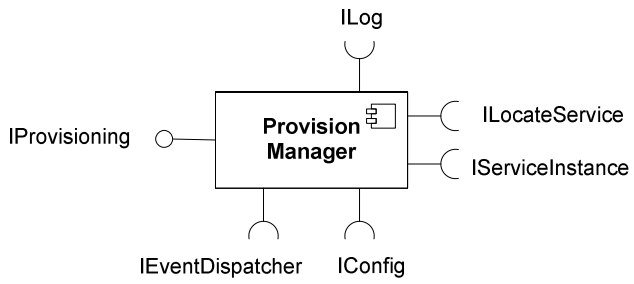

Figure 1 Provision Manager component

The client requests creation or deletion of a Subscriber by providing a Subscription.

3.2 Exported Interfaces

The Provision Manager exports the following interfaces: IProvisioning.

3.2.1 IProvisioning

The IProvisioning interface is used to manage Subscribers in the user registry. The user of this interface will be notified if a request fails and this notification will contain the CAI error codes.

3.2.1.1 Methods

3.2.1.1.1 Create subscriber

void create(Subscription sub, String adminuid, String adminpwd);

This method is used when a subscriber is to be created.

If mandatory attributes are missing (see ref. [1]) in the Subscription parameter, the request will fail and logged as an error.

3.2.1.1.2 Create subscriber asynchronously

int createAsync(Subscription sub, String adminuid, String adminpwd);

This method is the asynchronous version of create. The method returns a transaction ID used for identifying the asynchronous request.

3.2.1.1.3 Wait for the create asynchronous request

void create(int TransactionId);

Used for waiting for the asynchronous createAsync request to be finished. The asynchronous request status can be polled using the isFinished method.

3.2.1.1.4 Delete subscriber

void delete(Subscription sub, String adminuid, String adminpwd);

This method is used when a subscriber is to be deleted. Only the Telephonenumber in Subscription is taken into concern here.

3.2.1.1.5 Delete subscriber asynchronously

int deleteAsync(Subscription sub, String adminuid, String adminpwd);

This method is the asynchronous version of delete. The method returns a transaction ID used for identifying the asynchronous request.

3.2.1.1.6 Wait for the create asynchronous request

void delete(int TransactionId);

Used for waiting for the asynchronous deleteAsync request to be finished. The asynchronous request status can be polled using the isFinished method.

3.2.1.1.7 Check if request is finished.

boolean isFinished(int transactionID);
Returns the status of the asynchronous request identified by transactionID. If the request is finished it returns true, otherwise false.

3.2.1.2 Parameter type description

3.2.1.2.1 Subscription

The Subscription parameter contains list of attribute-value pairs. The format of the Attribute name and the value is as used in the CAI interface.

3.2.1.2.2 Adminuid

This parameter contains a valid uid of the provisioning administrator to use. Valid means that it must exist in the user register.

3.2.1.2.3 Adminpwd

This parameter contains the password of the provisioning administrator to use.

3.3 Imported Interfaces

The Provision Manager uses the following external interfaces:

- ILog for logging
- IConfig for configuration
- ILocateService for M3 service lookup
- IServiceInstance for accessing service properties
- IEventDispatcher for retrieving events

3.4 Functions

3.4.1 CAI communication

The Provision Manager communicates with the CAI server using the CAI interface, see ref. [1]. The communication uses a configurable number of connections (using a connection pool). The lifetime of a connection is configurable. The CAI Server may disconnect an established connection for example if it has been idle for some time. This connection will be re-established by Provision Manager the next time it will be used. A new pool is created for each unique provisioning administrator used.

The Provision Manager sets a connection timeout for the CAI connections to avoid connections that will hang forever if the CAI server is down.

If a CAI command fails, the Provision Manager can re-try the command. The decision whether or not to retry the command depends upon how many retries that have been configured.

3.4.2 CAI server Host

The Provision Manager retrieves the CAI server to communicate with using the ILocateService interface. The Provision Manager asks the component register for a Service Instance and it is the component register that is responsible for determining which instance that is appropriate.

When Provision Manager has got an instance it will keep to that instance as long as it is working. I.e. there will be no load balancing between different CAI servers.

3.4.3 Subscription attributes

The possible attributes (included in the subscription parameter) to use when creating a subscriber are defined in ref. [1]. Some of those attributes are mandatory. In case they are not set by the client, the Provision Manager may populate them before the request is sent to the CAI server. Currently only the mailhost attribute will be added by the Provision Manager in case it is missing. A configured value could be used for this purpose.

The Provision Manger will not in any other aspects try to validate the attributes in the Subscription. Instead the CAI server will do all the necessary control and if that validation fails it will respond with an error response.

3.4.4 Creation of a subscriber, first request

This scenario shows how the Provision Manager creates a subscriber the very first time in the life time of the Provision Manager.

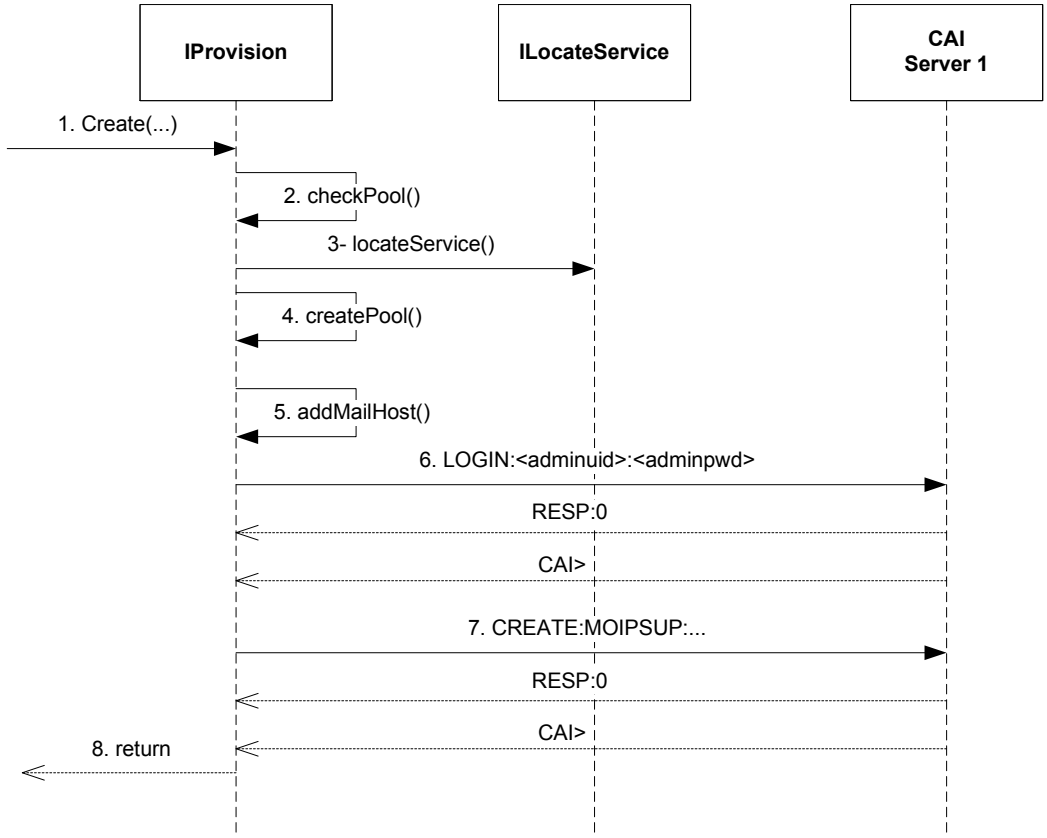

Figure 2 Creation of a subscriber, first request

1. The client request creation of a subscriber providing a Subscription and provision administrator credentials.
2. The Provision Manager check if there exist any pooled connections for this provision administrator or any CAI server at all.
3. Because no CAI server was found, an instance is requested from the ILocateService interface. The service to use is configured in Provision Manager and defaults to “Provision”.
4. A new pool is created using this CAI server and the provision administrator credentials.
5. The Subscription in the create request is examined for the existence of the Mailhost attribute. If it is missing, a configured Mailhost is added to the Subscription.
6. The Provision Manager logs into the CAI server using the provision administrator credentials
7. The subscriber is created using all applicable attributes from the Subscription.
8. The CAI result code is returned to Client.

3.4.5 Create using an established connection

This scenario shows how a request utilizes an already established connection to a CAI server.

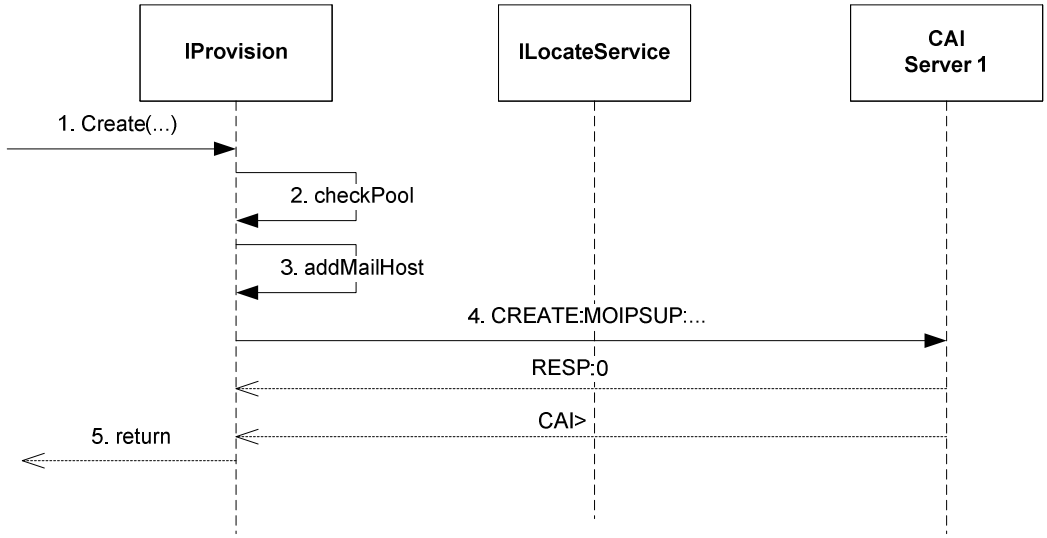

Figure 3 Create using an established conection

1. The client request creation of a subscriber providing a Subscription and provision administrator credentials.
2. The Provision Manager check if there exist any pooled connections for this provision administrator or any CAI server at all.
3. Because there was an existing connection available, the Provision Manager continues to check the Subscription in the create request for the existence of the Mailhost attribute. If it is missing, a configured Mailhost is added to the Subscription.
4. The subscriber is created using all applicable attributes from the Subscription.
5. The CAI result code is returned to Client.

3.4.6 Asynchronous/Synchronous mode

The methods create and delete defined in the interfaces in chapter 3.2 does also exist in asynchronous versions. The asynchronous functionality is described in Figure 4.

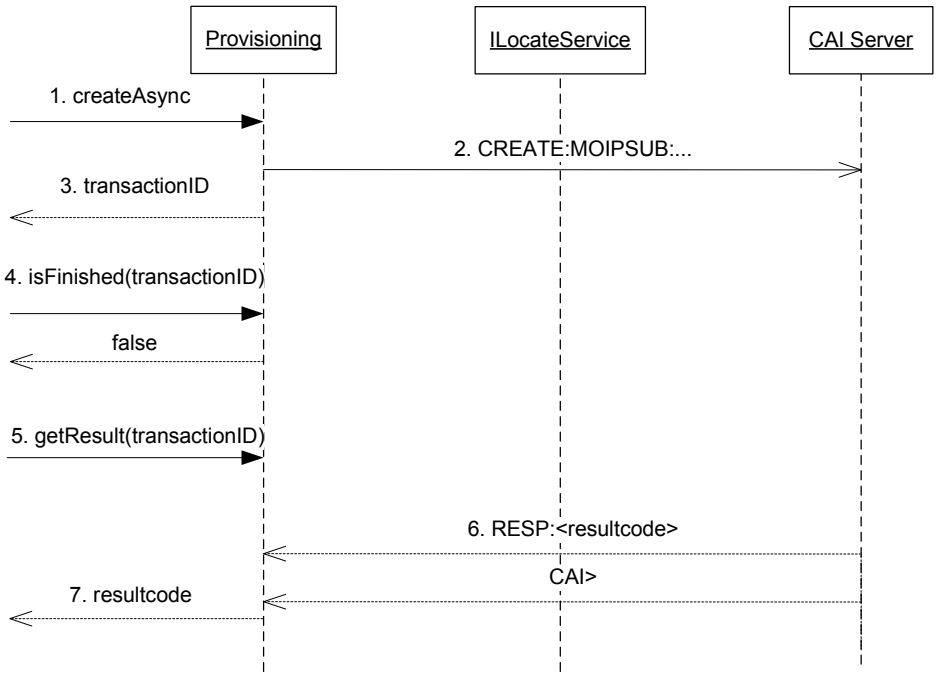

Figure 4 An asynchronous request

1. A subscriber creation is asynchronously requested.
2. The CAI request is sent to the CAI server host.
3. A transaction ID for the creation request is returned for later queries.
4. The Provision Manager can be polled about the status of the request.
5. The result is requested. If the creation is finished, the result will be returned immediately. Otherwise, it will wait for the CAI request to finish (or timeout) before returning.
6. The result from the CAI server arrives eventually
7. The CAI result code is returned to the client.

3.4.7 Fault scenarios

3.4.7.1 CAI server errors

When communication with a CAI server fails, the Provision Manager reports this to the Component Register and retries the search request, see Figure 5.

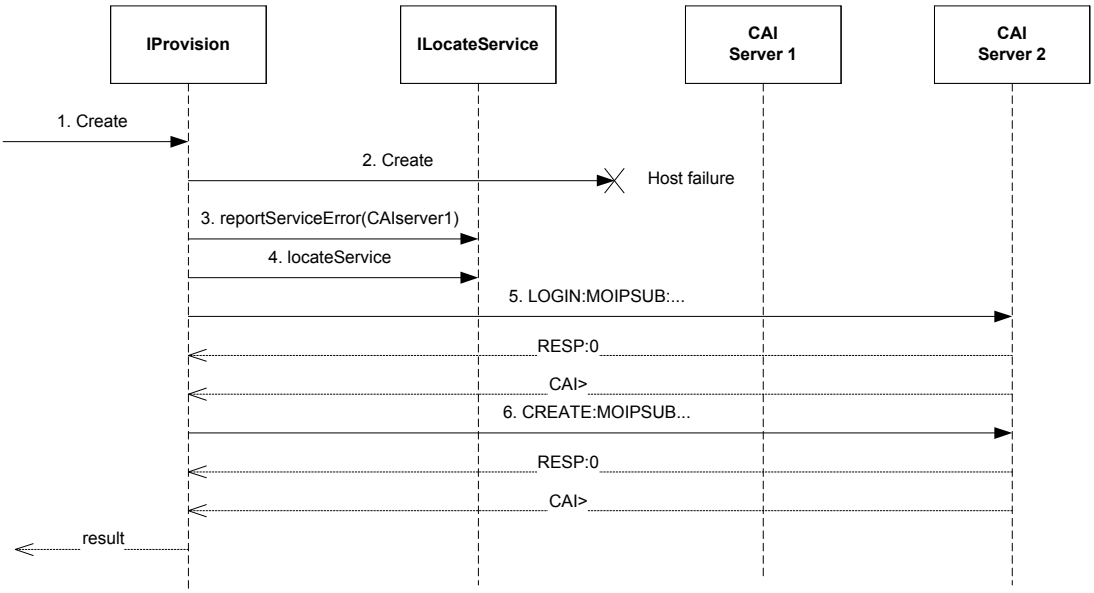

Figure 5 Host failure

1. A create is requested.
2. The CAI request fails due to communication errors with the CAI server.
3. The host failure is reported using the ILocateService interface.
4. Another CAI Service Instance is retrieved using the ILocateService interface.
5. A login is performed to the new CAI server
6. The request is performed once again
7. The result is returned back to the client.
8. The sequence 4 through 7 is performed a maximum, configurable number of times.

3.4.7.2 CAI Error codes

In case CAI responds with something else than success (code 0) the following scenario applies. If the CAI error code indicates that the fault is temporarily, the Provision Manager retries the command a configured number of times. Otherwise the client to Provision Manager is notified that the request failed including the CAI error code.

3.4.7.3 No CAI response

If the CAI server never returns a response on a request, the Provision Manager waits a maximum, configurable time and then treats this request as failed. In this case it throws an exception indicating timed out.

3.4.7.4 No CAI servers found

If the Provision Manager cannot get any CAI Service Instances using the ILocateService interface, it throws an exception indicating service not available.

3.4.8 Events

3.4.8.1 Consumed

The Provision Manager registers itself using the IEventDispatcher interface as a receiver of the following events:

- Configuration has changed (indicates that configuration should be re-read)
- Service Not Available (indicates that a host has been removed from MCR)

3.4.8.2 Produced

- The Provision Manager does not produce any events.

4 External Operation Conditions

4.1 Configuration

The following are configurable in the Provision Manager:

- Size of a connection pool.
- Idle timeout for a connection, i.e. how long the connection can be unused before it is considered idle and is disconnected.
- Timeouts on when a connection is created
- Number of retries in case of communication error
- Number of retries in case of CAI error responses
- Mailhost if missing in the create request.
- The maximum time to wait for a response from the CAI server before the request is assumed failed.

4.2 Logging

- Hosted service circumstances are logged as described in ref.[4] using the IHostedServiceLogger interface.
- All calls to interface methods are logged at info level. Method name and parameter values are logged. At return returned value is logged.

5 Capabilities

This paragraph is intentionally left blank.

6 References

[1] IWD Provisioning Interface
2/155 19-CRH 109 086 Uen

[2] FS External Component Register
9/FS-MAS0001 Uen

[3] FS Operate and Maintain Manager
17/FS-MAS0001 Uen

[4] FS-Log Manager
2/FS-MAS0001 Uen

7 Terminology

CAI	Customer Administration Interface