NOKIA

InstantLink
Release 19

Functional Description

Disclaimer

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

The information in this document applies solely to the hardware/software product ("Product") specified herein, and only as specified herein. Reference to "Nokia" later in this document shall mean the respective company within Nokia Group of Companies with whom you have entered into the Agreement (as defined below).

This document is intended for use by Nokia's customers ("You") only, and it may not be used except for the purposes defined in the agreement between You and Nokia ("Agreement") under which this document is distributed. No part of this document may be used, copied, reproduced, modified or transmitted in any form or means without the prior written permission of Nokia. If You have not entered into an Agreement applicable to the Product, or if that Agreement has expired or has been terminated, You may not use this document in any manner and You are obliged to return it to Nokia and destroy or delete any copies thereof.

The document has been prepared to be used by professional and properly trained personnel, and You assume full responsibility when using it. Nokia welcomes your comments as part of the process of continuous development and improvement of the documentation. This document and its contents are provided as a convenience to You. Any information or statements concerning the suitability, capacity, fitness for purpose or performance of the Product are given solely on an "as is" and "as available" basis in this document, and Nokia reserves the right to change any such information and statements without notice. Nokia has made all reasonable efforts to ensure that the content of this document is adequate and free of material errors and omissions, and Nokia will correct errors that You identify in this document. Nokia's total liability for any errors in the document is strictly limited to the correction of such error(s). Nokia does not warrant that the use of the software in the Product will be uninterrupted or error-free.

NO WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF AVAILABILITY, ACCURACY, RELIABILITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, IS MADE IN RELATION TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT WILL NOKIA BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT, EVEN IN THE CASE OF ERRORS IN OR OMISSIONS FROM THIS DOCUMENT OR ITS CONTENT.

This document is Nokia proprietary and confidential information, which may not be distributed or disclosed to any third parties without the prior written consent of Nokia.

Nokia is a registered trademark of Nokia Corporation. Other product names mentioned in this document may be trademarks of their respective owners.

Important Notice on Product Safety

This product may present safety risks due to laser, electricity, heat, and other sources of danger.

Only trained and qualified personnel may install, operate, maintain or otherwise handle this product and only after having carefully read the safety information applicable to this product.

The safety information is provided in the Safety Information section in the "Legal, Safety and Environmental Information" part of this document or documentation set.

Nokia is continually striving to reduce the adverse environmental effects of its products and services. We would like to encourage you as our customers and users to join us in working towards a cleaner, safer environment. Please recycle product packaging and follow the recommendations for power use and proper disposal of our products and their components.

If you should have questions regarding our Environmental Policy or any of the environmental services we offer, please contact us at Nokia for any additional information.

1 About This Document

This document describes the InstantLink system and gives an overview of it. The document explains how InstantLink is positioned in the fixed, mobile, satellite, IP or data network and what functions the system offers to network operators.

1.1 Audience

This document is intended for those who are interested in an overview of the InstantLink system and its functionality.

1.2 Terms and Concepts

The following abbreviations, terms and concepts are used in the document:

1.2.1 Abbreviations

API	Application Protocol Interface
AUC	Authentication Centre
CORBA	Common Object Request Broker Architecture
DSL	Digital Subscriber Line
FIFO	First-In-First-Out
GRC	Global Resource Configuration
HLR	Home Location Register
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP Secure
IIOP	Internet Inter-ORB Protocol
IMSI	International Mobile Subscriber Identity
I/O	Input/Output
IP	Internet Protocol
ISM	Intermediate Session Management
LDAP	Lightweight Directory Access Protocol
LTT	InstantLink Logic Testing Tool
MML	Man Machine Language
MMSC	Multimedia Messaging Service Centre
NE	Network Element
NEI	Network Element Interface
Nemo	Network Model Manager
OSS/BSS	Operations and Business Support System
Q3	Standard management interface for telecommunications management network.
RMI	Remote Method Invocation
RPE	Request Processing Engine
SME	Service Module Engine
SSL	Secure Sockets Layer
SSO	Single Sign-On
TE	Task Engine
TLS	Transport Layer Security
UI	User Interface
VNE	Virtual Network Element
XML	Extensible Markup Language

1.2.2 Terminology

business request	See request.
distribution	Running request and task processing components on more than one host to have better use of available physical resources.
network element; NE	A piece of communications equipment, such as a switch, a Home Location Register or a Voice Mail System that provides support or services to the subscriber.
network model	A service provisioning view to the operator's provisioned network elements. A network model consists of network elements and connections between InstantLink and network elements.
notification	A message with the status of a request that InstantLink creates for the OSS/BSS during the execution of a long-lasting request. Notifications are typically defined in provisioning workflow and the number of notifications is not limited. A notification contains the request identifier, request status and a free format notification message.
OpenAM	An opensource middleware software component that provides authentication, authorization, entitlement and federation functionalities.
Operations and Business Support System; OSS/BSS	A program that helps an operator monitor, control, analyse and manage usage of a communications network. OSS/BSS systems include, for example, systems for customer care, order management, billing, relationship management, decision support, market analysis, fraud detection, traffic engineering and network planning. The individual properties of each OSS/BSS system determine what kind of requirements the customer sets for the Provisioning Systems, such as what services the Provisioning System should activate for subscribers.
organisation	An internal party of an operator, for example, prepaid division or broadband division, or an external party such as a virtual network operator that is accessing InstantLink. The system administrator (network owner) can limit the access of an organisation to a subset of InstantLink functionalities. Operators can define network elements, provisioning logics, bandwidth and number ranges for the organisations and create client users and UI users for each organisation with different levels of access rights within the organisation.
Request Processing Engine	An InstantLink component that manages requests, ensures correct request execution order and acts as a load balancer.
request	A work order that InstantLink receives from an OSS/BSS (or Provisioning Client) that consists of a required operation, which is aimed at one subscriber in one or more network elements. A request consists of required operations, such as create, delete, query and modify subscriber data, which are aimed at a subscriber in one or more network elements. The InstantLink system processes requests according to configured rules. The result is a set of executable tasks.

response	A message that contains the completion status of a provisioning request. InstantLink creates a response message for each provisioning request after the request has been completed. The response provides the OSS/BSS with the necessary information about the request, such as the request identifier and information on whether the execution has succeeded.
service module	A module that implements additional request processing functionality to InstantLink, for example, Business Service Tool and InstantLink Routing Service Module.
Service Module Engine	An InstantLink component that ensures task generation based on provisioning logic's workflow. This component can be distributed over several hosts.
session	A communication channel reserved for a certain period of time.
task	A subscriber or service management operation generated from a request and affecting one network element.
Task Engine	An InstantLink component that manages tasks. This component can be distributed over several hosts.
virtual network element; VNE	A symbolic network element that can represent several network elements of the same type in the network model. The virtual network element can be used when InstantLink is accessing a large number of network elements that are managed in some other system, typically in a Network Inventory product (for example, Inventory). Instead of storing all network element information into the InstantLink network model, only virtual network element information is stored and the exact information of the physical network elements is retrieved from the network inventory system.

1.3 Related Documentation

For more information on InstantLink, see the following documents:

- InstantLink Release Notes
- InstantLink Operation and Maintenance Guide
- InstantLink Reference Manual
- InstantLink Online Help
- Provisioning and Activation Installation Guide

For more information on other Provisioning and Activation components, see the component-specific documentation.

2 Introduction

All subscribers and services that are defined in the Operations and Business Support Systems (OSS/BSS) have to be configured in the appropriate network elements before customers can use their services. Provisioning refers to transmitting this subscriber and service information from the OSS/BSS to the network element.

Figure 1 illustrates provisioning on a general level.

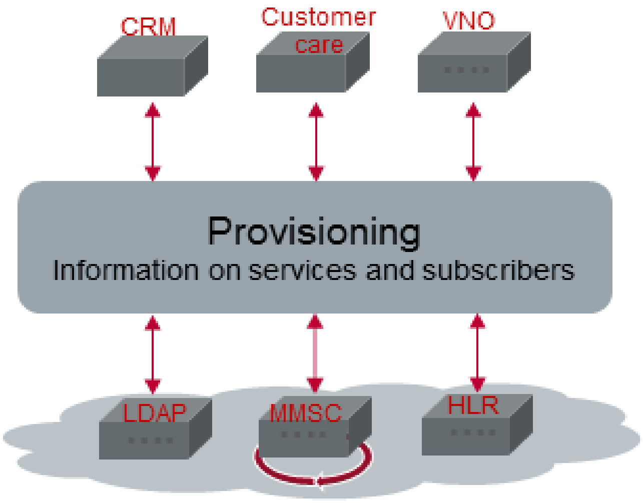


Figure 1. Provisioning

In the operator’s network, InstantLink conveys information on services and subscribers between OSS/BSS systems and network elements. For more information on how InstantLink is positioned in the operator’s network, see Section 2.1 *Environment*.

InstantLink’s architecture consists of an OSS/BSS interface, Request Processing Engine (RPE), Service Module Engines (SMEs), Task Engines (TEs), network element interfaces (NEIs), Network Model Manager (Nemo) and database. For more information on the InstantLink architecture, see Chapter 3 *Architecture*.

InstantLink is a provisioning system through which a service provider can create subscribers, modify and query subscriber data, activate new services to existing subscribers, and delete subscribers and their services. InstantLink automates the provisioning and activating of services in fixed, mobile, satellite, IP and data networks, so that no manual intervention is needed from the operator's personnel. OSS/BSS systems do not need to have profound knowledge of the network itself; InstantLink takes care of the communication with network elements. For more information on InstantLink functionality, see Chapter 4 *Functionality*.

2.1 Environment

InstantLink is a provisioning platform between the OSS/BSS systems and the telecommunication network. It provides a single point of connection for the OSS/BSS systems to manage subscriber and service information in different network domains.

InstantLink can receive activation orders from any OSS/BSS, such as customer care systems or self-service platforms, and can be used in a variety of different network domains, such as 3G, IP, IMS, WLAN, PSTN, cable and satellite.

InstantLink has interfaces with various OSS/BSS systems and network element types. InstantLink communicates with OSS/BSS systems through OSS/BSS interfaces and with network elements through network element interfaces. Typical network elements in a mobile operator include Home Location Register (HLR), Authentication Centre (AUC) or Lightweight Directory Access Protocol (LDAP). For a broadband operator, InstantLink processes requests into network inventory, element management systems and value-adding service elements such as e-mail, VoIP and IPTV.

Figure 2 illustrates the position of the InstantLink system in the operator’s network.

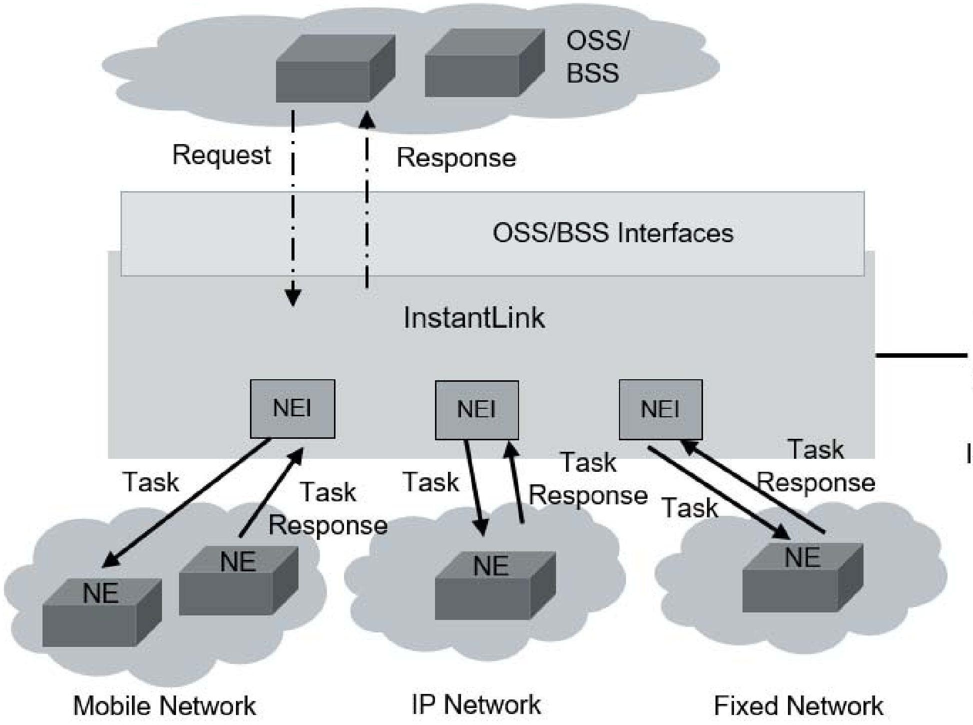


Figure 2. InstantLink in the operator’s network

InstantLink provides open interfaces for OSS/BSS systems to provision subscribers. It activates services using configurable processing rules into a heterogeneous network that has different types of interfaces.

An OSS/BSS sends requests to and receives responses from InstantLink through OSS/BSS interfaces. InstantLink can support several OSS/BSS systems and OSS/BSS interfaces simultaneously.

InstantLink sends the commands generated from the requests to the network elements across the network element interfaces (NEIs). Also, the network elements send response data to InstantLink across these interfaces. InstantLink can support several network element types and versions simultaneously.

3 Architecture

The InstantLink system processes requests in several components that communicate with each other. An OSS/BSS interface is a communicator between OSS/BSS systems and Request Processing Engine, and it changes requests into InstantLink internal format. Request Processing Engine handles requests from and delivers responses to OSS/BSS systems, handles request locking and acts as a load balancer. Service Module Engine provides the scalability in the InstantLink architecture. It holds the service modules and delivers tasks to Task Engine(s).

Figure 3 illustrates the architecture and components of InstantLink.

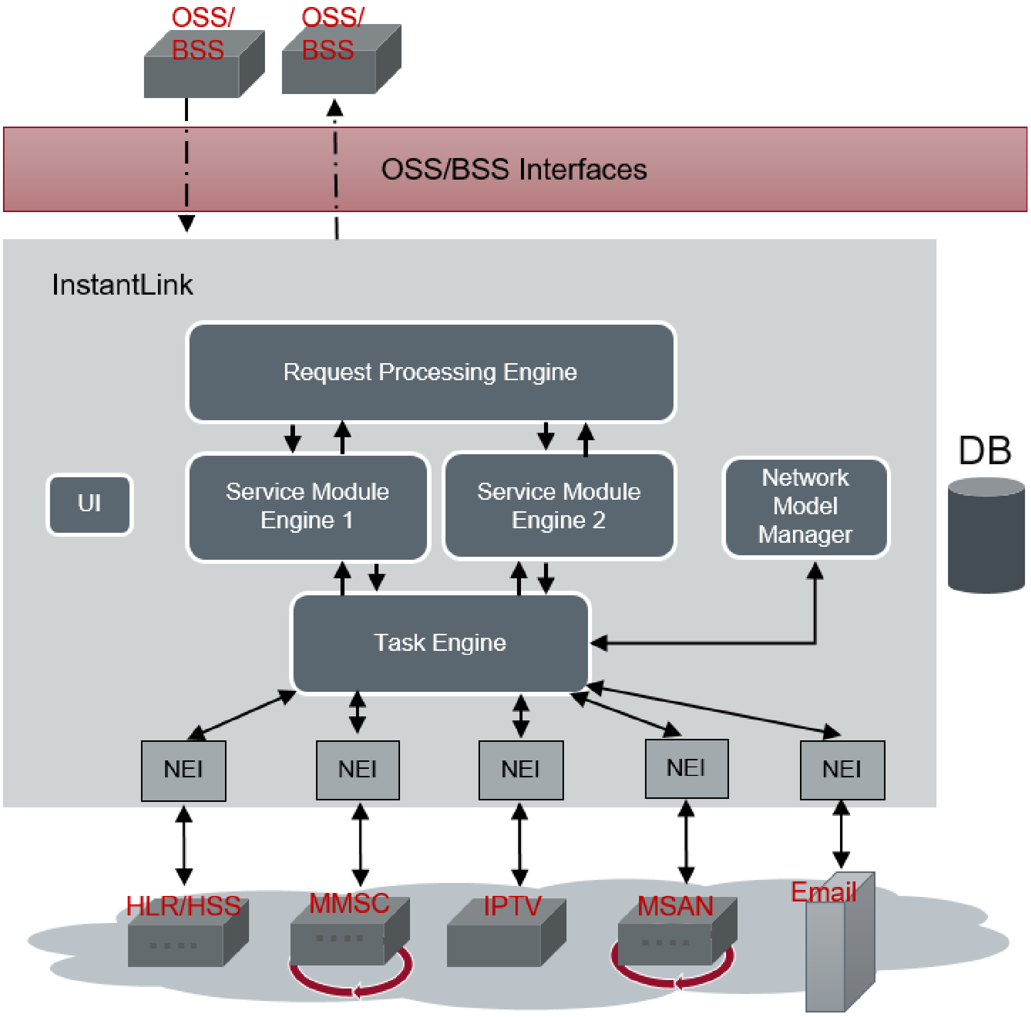


Figure 3. InstantLink system architecture

InstantLink is composed of the following components:

- OSS/BSS interface
- Request Processing Engine
- Service Module Engine

- Task Engine
- Network Model Manager
- Database

By default, the InstantLink installtion includes one Request Processing Engine, Network Model Manager, Task Engine and two Service Module Engines.

Request Processing Engine acts as a central point of contact to all InstantLink components. All components register themselves to Request Processing Engine when they start up. Request Processing Engine divides the request load between the Service Module Engines based on a load balancing algorithm. The load balancing functionality ensures that all Service Module Engines are efficiently utilised.

Network Element Interfaces communicate with network elements. Network Model Manager is responsible for maintaining the provisioned network topology and status information.

The components of InstantLink are presented in more detail in the following sections.

3.1 Request Processing Engine

Request Processing Engine receives provisioning requests from OSS/BSS systems, loads unsent requests from the database during startup, generates responses for the requests and delivers the responses to the OSS/BSS systems. Request Processing Engine also handles request locking and takes care of load balancing between Service Module Engines.

3.1.1 Request Processing

Request Processing Engine serves as an interface towards OSS/BSS systems and manages connections to them. Request Processing Engine manages the transactions to two directions: it receives requests and sends responses.

An OSS/BSS opens a connection for sending requests or receiving responses by sending a login message to InstantLink. When Request Processing Engine receives the login message, it sends an acknowledgment to the OSS/BSS and a connection is opened. A connection stays open until the OSS/BSS closes it, or the InstantLink system is shut down. If for some reason the login message is not accepted and the OSS/BSS is not authenticated, Request Processing Engine sends an error message to the OSS/BSS and does not open a connection.

Request Processing Engine validates the received requests, converts them into an internal format, saves them to the database and gives them a unique identification. It creates a request execution queue, which holds the execution order according to the priority of the request and the requested execution time. Request Processing Engine also keeps a lock list that ensures that there are no parallel requests for the same subscriber executed at the same time.

Request Processing Engine searches the execution queue for new executable requests. It always starts from the beginning of the queue and takes the first request that can be executed. Then it checks the lock list to see if the request can be executed. If the request can be executed, it is sent to a Service Module Engine according to a load balancing algorithm.

Request Processing Engine receives responses from the Service Module Engine and removes the request locking identifier from the lock list. It then converts the responses into the correct protocol format and sends them to the OSS/BSS.

Request Processing Engine provides open interfaces for OSS/BSS integration, namely Extended API. In addition, InstantLink supports other vendor-specific and industry-standard interfaces to OSS/BSS systems, such as InstantLink SOA Web Services, Batch API and Edifact API. For more information about the Extended API protocols, see *InstantLink Extended API Functional Description*.

3.1.2 Load Balancing

Request Processing Engine divides the request load between the Service Module Engines based on a load balancing algorithm. The load balancing algorithm takes into account the load of the Service Module Engines and divides the request load so that all Service Module Engines are efficiently utilised. For more information on load balancing algorithms, see *InstantLink Online Help*.

3.2 Service Module Engine

Service Module Engine derives tasks from the requests received from Request Processing Engine and sends the tasks to Task Engine. Service Module Engine then receives responses to the tasks from Task Engine.

Service Module Engine offers an interface for plug-in service modules that can be used for:

- routing tasks to network elements (InstantLink Routing Service Module)
- executing a provisioning logic (Business Service Tool)
- simulating network element responses when testing a provisioning logic (InstantLink Logic Testing Tool)

For more information on service modules, see Section 4.10 *Service Modules*.

Service modules can be turned on and off in the InstantLink user interface. They each have their own user interface, which is integrated to the InstantLink user interface.

Figure 4 illustrates the service modules of InstantLink.

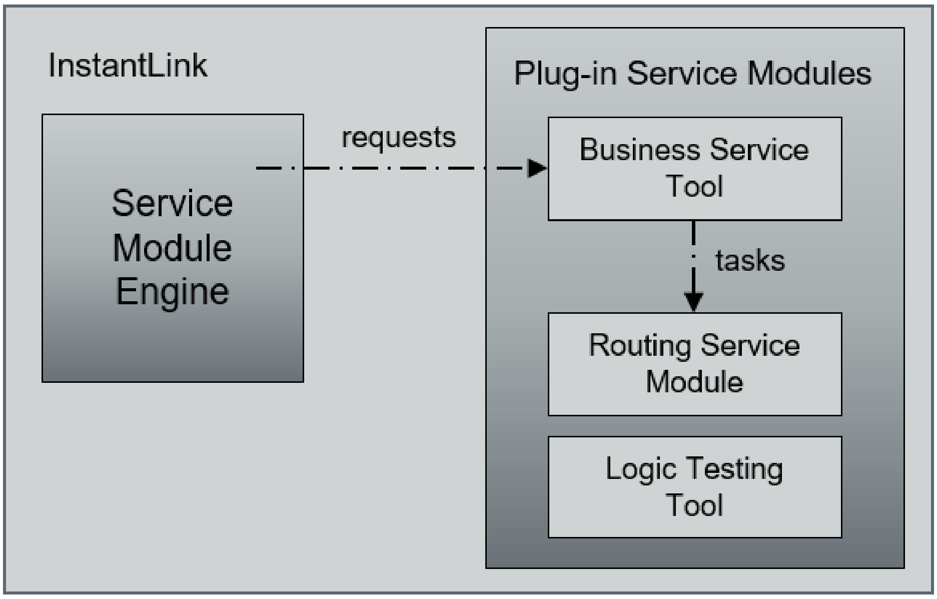


Figure 4. Service modules

In the figure above, Service Module Engine sends the requests it receives from Request Processing Engine to Business Service Tool where requests are divided into tasks. Then Business Service Tool sends the tasks to Service Module Engine and, if a suitable routing condition is found, network element identification is added to the task.

Service Module Engine sends the tasks to Task Engine. It receives responses to the tasks from Task Engine, collects all the task responses concerning one request and converts them into request responses. Service Module Engine then sends the request responses to Request Processing Engine.

Task Engine informs Service Module Engine of the available network elements. Service Module Engine stores the tasks that cannot be sent to Task Engine, for example, if a network element is not reachable. When Task Engine allows task sending to the previously blocked element, the unsent tasks are fetched from the database and sent to Task Engine. If the network element does not become available before the task expires, a message is sent back to Service Module Engine that the task was failed.

If there is a connection error, Service Module Engine selects the correct backup network element, if available, and starts sending tasks to the backup network element until the default network element is recovered.

Service Module Engine enables sending timed tasks to Task Engine which forwards them to the network elements. Service Module Engine also queries the identifications of all network elements from Task Engine after startup and system refresh.

3.3 Task Engine

Task Engine receives tasks from Service Module Engine and puts them into task queues to be sent to specific network elements.

The task queue has a configurable size that holds tasks that are about to be forwarded to a network element. There is one task queue for each managed network element in the network model. Tasks are taken out of the queue in First-In-First-Out (FIFO) order. When the queue is full, new tasks are not accepted until there is space available again.

Task Engine allocates network resources from the network model and assigns them to sessions. Task Engine uses sessions for sending tasks for execution. For a session, Task Engine needs a route consisting of NEs and connections from Network Model Manager (for more information, see Section 3.4 *Network Model Manager*). With the help of these resources, Task Engine can connect to a network element and process tasks. There can be several concurrent connections to one network element.

Task Engine allows receiving tasks including all IP addresses that belong to the same virtual network element (VNE) concurrently and in a controlled manner. When VNEs are used, Task Engine first receives data from Network Model Manager and checks whether tasks are to be directed to VNEs, and if they are, it then forwards tasks to the proper VNEs.

Task Engine sends responses to Service Module Engine. If for some reason task responses cannot be sent to Service Module Engine, the responses are stored in a file. When Service Module Engine is able to receive task responses again, Task Engine sends the stored responses.

3.4 Network Model Manager

Network Model Manager (Nemo) is responsible for maintaining the provisioned network topology and status information. Network Model Manager informs Task Engine when a network element connection is down and takes care of sending tasks to the backup network element until the primary network element is recovered.

Network Model Manager maintains and updates information about:

- network elements
- connections between the network elements
- statuses of the network elements and connections

The network model is defined and maintained through the InstantLink user interface. There is one Network Model Manager per InstantLink environment.

3.5 Database

InstantLink stores information on network elements, requests and system and distribution configuration to its own database. InstantLink uses an Oracle database. There is one database for the distributed InstantLink environment.

The InstantLink database includes information on user privileges and access rights. InstantLink has the UI and client user passwords stored into the database in a SHA-256 hashed form, which promotes system security. Southbound interface passwords are encrypted with AES-128. Sensitive request parameters can be encrypted while processed within InstantLink. This feature is configurable and encryption can be done with 3DES or AES-128 encryption algorithms.

In addition to the main InstantLink database, the system can be extended to include Archive Database which can be used for storing and monitoring old requests.

InstantLink offers database cleaning tools which enable fast and efficient database maintenance procedures.

4 Functionality

This chapter describes the process in which requests pass through the InstantLink system.

4.1 Provisioning Functionality

InstantLink functionality is based on processing requests and tasks in the network.

InstantLink provisioning functionality consists of:

- processing requests (standard and LITE requests, S-LITE tasks)
- processing tasks (synchronous and asynchronous)
- sending responses
- locking requests to ensure that only one request focusing on a subscriber is executed at a time
- scheduling and prioritising requests
- resending requests and tasks
- collecting data on request and task statuses
- controlling intermediate sessions
- supporting virtual network operators
- managing user interface user sessions
- network distribution

InstantLink receives requests from OSS/BSS systems to activate services in the network. Once InstantLink has received a request, it finds the target network elements in which the service needs to be activated. InstantLink uses separately configurable service modules (for more information, see Section 4.10 *Service Modules*) for translating the request to network element specific operations, called tasks, and sends these tasks to the network element.

There are two kind of requests, standard and LITE. LITE requests are used for time critical operations, and a response is sent within a given time. LITE requests are more suited to data query operations where a record of their execution is not essential.

InstantLink stores all data on the provisioning operations in its database, with the exception of LITE requests which are not stored to the database. This enables faster request processing compared to standard requests. Statistics are collected for both LITE and standard requests. The user can monitor request and task processing through the InstantLink user interface.

InstantLink prepares a response for each completed request and delivers the response to the OSS/BSS that sent the request. The response informs the OSS/BSS of how the execution of the request succeeded.

When the OSS/BSS wants to activate several services concerning the same subscriber, InstantLink guarantees that these operations are performed in the order in which the OSS/BSS entered them. The operator can define how InstantLink proceeds if an error occurs while executing a service order through Business Service Tool provisioning logic rules.

InstantLink runs in the background and does not require any specific user interaction. However, the system administrator can monitor the provisioning activities, manage the InstantLink processing, and configure new network elements into the system through the InstantLink user interface.

InstantLink provisioning functionality is presented in more detail in the following sections.

4.1.1 Processing Requests and Tasks

A request is a work order including subscriber or service information that InstantLink receives from the OSS/BSS.

A request contains:

- requested provisioning operation (such as subscriber creation, service activation or data query)
- subscriber identification (such as a telephone number or IMSI)
- service identification (such as a Digital Subscriber Line (DSL), call forwarding or subscriber authentication)
- service parameters (such as call forwarding numbers or customer address)
- a request identifier supplied by an OSS/BSS
- optional request category: standard (default) or LITE

A request consists of one or several tasks. A task is a provisioning operation concerning subscriber data in one network element. When InstantLink receives a request from an OSS/BSS, it converts the request into a single task to be executed in the corresponding network element or uses Business Service Tool to decompose the request into multiple network level tasks according to provisioning logic rules.

Standard requests can also have a type of light-weight, faster executing task called an S-LITE task. S-LITE tasks have minimal interaction with the database and can be used for tasks that do not change the network, for example, display tasks. S-LITE tasks provide a sizable performance gain. For more information on S-LITE tasks, see *InstantLink Reference Manual*.

Once the InstantLink system has completed a standard request, it creates a response and sends it to the OSS/BSS. Responses for LITE requests follow the same pattern, except in case of LITE request time-out, when a response with a negative status is sent immediately to the OSS/BSS and processing of the request stops. The response indicates how the request execution succeeded. The response also includes a request identifier that the OSS/BSS needs in order to recognise the corresponding request. Moreover, a response includes network element data, such as the services that were activated for a subscriber.

Figure 5 illustrates the processing of requests, tasks and responses in the system.

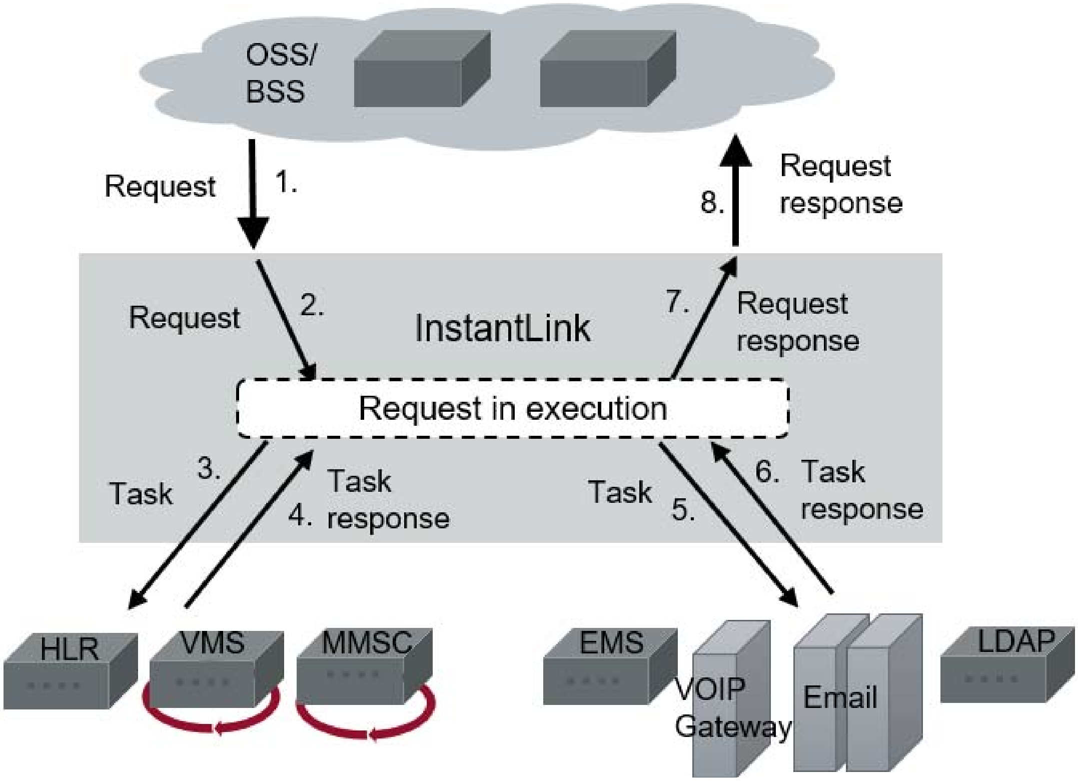


Figure 5. Requests, tasks and responses in InstantLink

As *Figure 5* illustrates, requests, tasks and responses are processed in the following order:

1. An OSS/BSS system sends a request to InstantLink.
2. InstantLink processes the request into one or more tasks. Business Service Tool can be used for request decomposition into multiple network element tasks.
3. InstantLink executes a task in a network element.
4. The network element sends a task response to InstantLink.
5. InstantLink executes another task in a network element.

6. The network element sends another task response to InstantLink.

7. When all tasks from one request are completed, InstantLink combines the task responses into one request response.

8. InstantLink sends the request response back to the OSS/BSS system.

InstantLink stores all requests and tasks in its database for monitoring and troubleshooting.

4.1.2 Locking Requests to Ensure Service Consistency on the Network Layer

To verify that requests are executed in the correct order, InstantLink supports locking requests that are under execution. Locking rules are used for verifying that requests are executed in the correct order. For instance, services for a single subscriber are activated first in HLR and then in AUC. These rules enable the user to 'lock' a request under execution so that no other requests for the same subscriber are executed before the processing of the locked request has reached its end status. Request locking is not available for LITE requests.

This feature ensures that there is always only one request at a time under execution for each subscriber. Although locking is in use, there can still be several requests and tasks processed in parallel in the system in case they do not have common key data, for example, a subscriber identifier. Locking rules are usually specified at installation and can be later modified.

For more information on locking rules, see *InstantLink Operation and Maintenance Guide*.

4.1.3 Scheduling and Prioritising Requests

In InstantLink, requests are normally executed instantly when they are received from the OSS/BSS interface. LITE requests are processed in preference to standard requests, regardless of request priority. There are two alternative ways to alter the normal execution pattern for standard requests: scheduling and prioritising.

Scheduling allows the user to define when a specified request is to be executed by InstantLink. InstantLink then starts executing the request at the predefined time instead of executing it immediately when the request is received.

Prioritising allows the OSS/BSS to send higher priority requests that pass the lower priority requests in execution queues.

Note There can be only 10 percent of requests or tasks with the highest priority in the entire traffic volume. Requests with the highest priority are threaded differently compared to lower priority requests. They bypass all the internal queues and are not buffered into the memory. A high load of priority 1 requests could exhaust the memory usage of the system.

For more information on scheduling and prioritising requests, see *InstantLink Operation and Maintenance Guide*.

4.1.4 Resending Requests

It is possible to resend a request through the InstantLink user interface, for example, when request sending fails because of long-term network connection problems. When a request is resent, InstantLink creates a new request based on the original request. A new request ID is given to the resent request.

4.1.5 Resending Tasks

It is possible to resend a task through the InstantLink user interface, for example, when task sending fails because of long-term network connection problems.

When a task is resent, InstantLink creates a new task based on the original task and adds the task in the original request. A new task ID is given to the resent task. Resent tasks always start from a task ID of 9000, because the maximum number of generated tasks in a request is 8999.

A resent task has two special features. Firstly, a resent task does not affect the final status of a request. Secondly, when a resent task is completed, InstantLink does not generate a response message.

4.1.6 Collecting Data on Request and Task Statuses

InstantLink collects and maintains data concerning request and task statuses. As requests and tasks are processed by the InstantLink system, they pass through various statuses. Since tasks are derived from requests, the request statuses depend on how the execution of the tasks has succeeded.

Statuses of standard requests and tasks are stored in the database and the InstantLink user can monitor them through the user interface. The user interface shows, for example, the reason for a failure in sending requests and tasks. This is important especially in troubleshooting.

LITE requests are not stored in the database in order to enable faster request processing. LITE requests cannot be viewed in the user interface.

S-LITE tasks are created in the database with one transaction and then, as long as execution proceeds quickly and successfully, they are never updated to the database again. In the user interface, their status is shown as `S-LITE` (this is purely cosmetic) although their underlying database status is `INITIATED` and it is not possible to search explicitly for `S-LITE` status tasks.

For more detailed information on request and task statuses, see *InstantLink Reference Manual*.

4.1.7 Notifications

Notifications provide intermediate progress information about long lasting requests. Notifications can be used, for example, to inform the OSS/BSS about major milestones, such as, a DSL line has been ordered from the network operator or a modem has been delivered to the end user. Notifications are sent to the OSS/BSS with the InstantLink SOA Web Services interface. Notifications can also be monitored through the InstantLink user interface.

InstantLink provides the following type of notifications:

- notifications sent from the Business Service Tool workflow
- notifications sent from network element interfaces (NEI phase messages)
- notifications generated automatically by InstantLink, for example, in case a request is in jeopardy

4.1.8 Controlling Intermediate Sessions to Network Elements

InstantLink provides a network element session control called Intermediate Session Management (ISM) for managing connections to network elements via an intermediate host. Intermediate Session Management ensures that all network elements are fairly served by sharing resources when they are limited.

Intermediate Session Management is only activated if enabled by a GRC parameter and if there are not enough connections from InstantLink to the intermediate host to serve all network elements and all their connections simultaneously. If activated, Intermediate Session Management periodically checks all task queues to network elements behind the intermediate host and controls the Task Engine sessions so that each task queue is given a fair allocation of resources. Intermediate Session Management decides the order in which resources are shared based on the priorities of the tasks queuing and the amount of time they have spent queuing. When there is no available connection for a particular network element, the network element’s status is set to waiting.

Figure 6 gives an example of Intermediate Session Management.

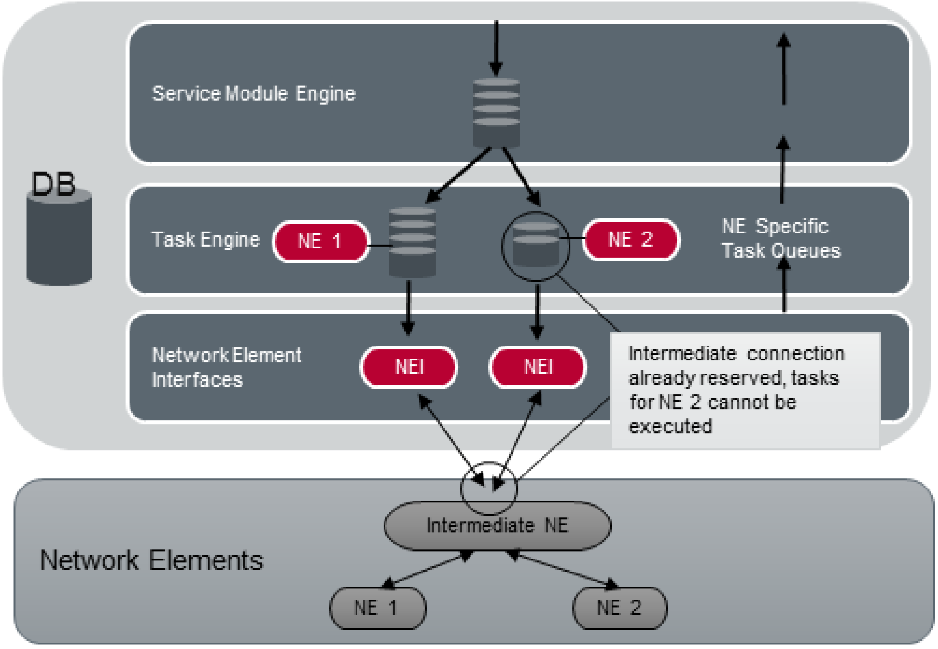


Figure 6. Example of Intermediate Session Management

4.1.9 Supporting Virtual Network Operators and Multiple Organisations

In InstantLink, it is possible to define organisations to support several virtual network operators within a single InstantLink instance. Operators can also have their own internal organisations. The network operator can define network elements, provisioning logics, bandwidth and number ranges for the organisations and create client users and UI users for each organisation with different levels of access rights within the organisation.

It is possible to define that client users that belong to an organisation can send tasks only to those network elements that are defined to be available for the organisation and only within the allowed subscriber identifier number range (if specified). Requests can be configured to be processed only by provisioning logics allowed for the organisation. UI users that belong to an organisation can also be restricted to monitor requests and tasks that belong to their organisation.

Organisation support can also be used for dividing provisioning to logical groups.

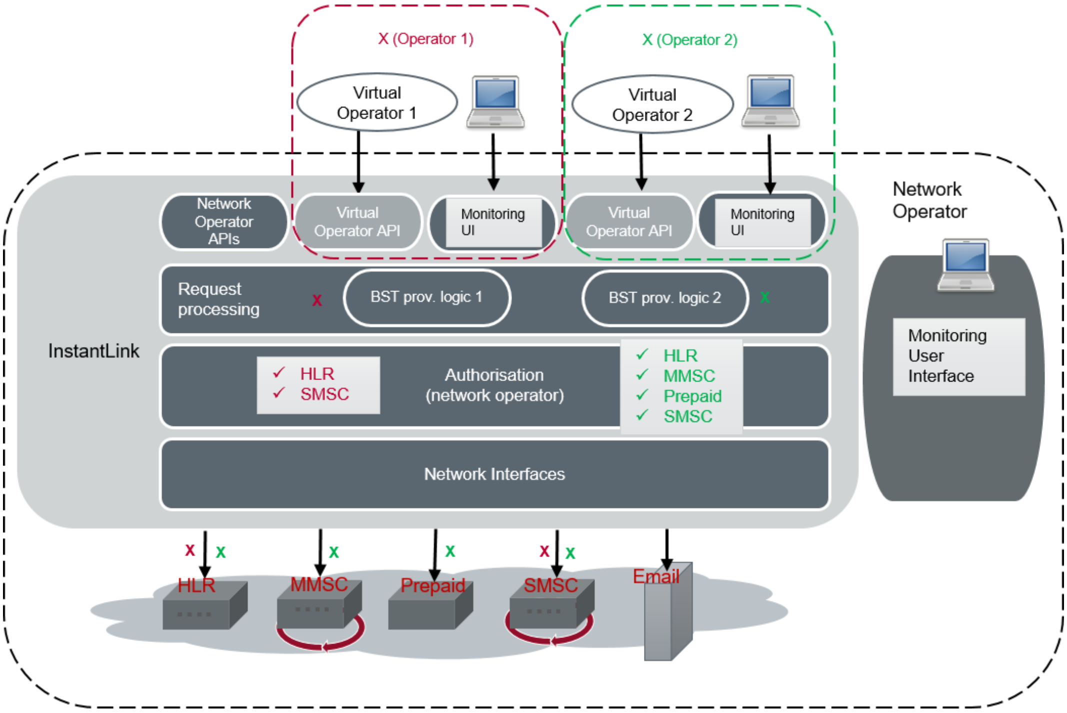


Figure 7. Virtual network operator support

Organisation support is visible in the user interface only when one or more organisations have been defined. For more information on the virtual network operators and organisations, see *InstantLink Online Help*.

InstantLink also allows allocating provisioning bandwidths for organisations. The provisioning bandwidth refers to the percentage of request throughput of InstantLink. For example, all client systems in one organisation share the same bandwidth. For more information on using provisioning bandwidths, see Section 4.6.1 *Provisioning Bandwidth to OSS/BSS Client Systems*.

4.1.10 Network Distribution

In a distributed InstantLink environment, the task load can be divided between several Task Engines on multiple hosts. There is one Task Engine per host. A network element's load is calculated based on the host load in the last 15 minutes, the average number of sessions in the last 30 minutes and the average number of received tasks in the last 30 minutes. For example, if one network element takes up a significant amount of load, the other network elements on the same host can be moved to another host with less load, so that all Task Engines get approximately an equal share.

Backup network element pairs are always handled by the same Task Engine. If the handling of a backup network element is moved to another host, its counterpart will also move. Intermediate network elements are also moved if they are connected to a backup network element pair that is moved to another host.

Network distribution is useful also if one Task Engine crashes. That Task Engine's network elements can then be divided between the remaining Task Engines until the crashed Task Engine is up again.

4.2 Startup, Shutdown and System Refresh

4.2.1 Startup

At startup, InstantLink reserves resources, such as database connections, reads its configuration and searches for executable requests from the database. InstantLink can also be configured to start other programs that are linked to the provisioning solution. For example, Workflow Client can be automatically started and shut down by the InstantLink startup and shutdown processes.

Task responses that were received by InstantLink are processed before the system is shut down. Unfinished requests are loaded from the database. After this, InstantLink accepts OSS/BSS connections and starts receiving new requests.

If some resources cannot be reserved, or some system layer is not able to start because of an error, InstantLink releases all the resources and shuts down.

4.2.2 Shutdown

After the user gives InstantLink the shutdown command, no new requests or tasks are taken into execution. The system waits for tasks that are sent to network elements for execution to be completed before shutting down.

The system does not wait for the completion of other active requests. Standard requests are stored in the database so that request processing may continue after InstantLink is restarted. Unfinished LITE requests are discarded and they are not stored in the database.

If an OSS/BSS tries to send requests to InstantLink when it is shutting down, InstantLink sends an error message to the OSS/BSS informing about the shutdown.

4.2.3 System Refresh

System refresh is used to take new system configuration into use. In system refresh, the system reads all the GRC parameters and other configurations again. Changes can affect the flow of requests and tasks, for example, when a new service module is added.

During system refresh, request processing stops for a few seconds. If a request is partly executed, the processing continues after the system refresh.

Note A break in request processing may cause a time-out for time critical LITE requests.

The configuration refresh does not affect the OSS/BSS system connections to InstantLink. All OSS/BSS connections remain active.

4.3 System Operation

This section briefly describes the system operations performed through the user interface.

Through the user interface the user can:

- monitor standard requests and tasks and their statuses in request processing
- view, manage and modify the network model
- resend requests and tasks
- create and manage InstantLink UI users
- create and manage OSS/BSS client users
- check the system status and statistics
- configure and manage service modules
- use testing tools for request generation and network element interface simulation
- manage distribution configuration

For more detailed descriptions, see *InstantLink Operation and Maintenance Guide* and *InstantLink Online Help*.

InstantLink supports the following languages in parameter value sending:

- Western European
- Simplified Chinese
- Thai

InstantLink uses the Western European character set by default. For more information on configuring another character set, see *InstantLink Character Set Configuration Guide*.

4.3.1 System Monitoring

Through the InstantLink user interface, users can monitor whether system components are running or not. Also connections between InstantLink and OSS/BSS systems or network elements can be monitored. Users can also view the volume of requests that are processed in the system. The configuration changes can be monitored from the audit trail.

If a user wants to open multiple sessions of the user interface in more than one browser, for example, to have different search criteria on the Requests page, the additional browsers must be opened from **Start** > **Programs**. If the additional browser page is opened as a new window from the browser, the browser considers it as the same session and the search criteria may be copied to another browser page unintentionally. When the sessions are separate, it is possible to browse multiple windows with one computer. Auto-refresh is also possible when the UI sessions are separate.

4.3.2 Distribution Configuration

The InstantLink user interface lists the hosts on which the InstantLink system has been installed and preconfigured. The hosts and their components can be modified through the InstantLink user interface. It is also possible to add new Service Module Engines to hosts. For more information, see *InstantLink Online Help*.

4.3.3 Log Files

InstantLink components write log files that contain detailed information on a component level about the system operation. Network element (NE) log files (MML logs, error logs and I/O logs) record communication between InstantLink and network elements. The log files can be accessed through the user interface or through UNIX command line.

NE logs older than one day can be compressed and transferred to a customer specified location(s) to be maintained more efficiently and archived. Archived NE logs can also be used when viewing request and task data from Archive Database. For more information, see *InstantLink Reference Manual*.

Log files can be cleaned manually or automatically. For more information, see *InstantLink Operation and Maintenance Guide*.

4.3.4 Statistics

InstantLink collects statistical information about requests and tasks as well as system functions.

InstantLink provides the following statistics:

- request and task statuses
- throughput statistics
- OSS/BSS statistics
- RMI FIFO statistics
- NE statistics
- load distribution
- processing history

For more information on statistical information in InstantLink, see *InstantLink Operation and Maintenance Guide*.

4.3.5 Reporting

InstantLink provides a number of built-in reports that can be viewed through the InstantLink user interface. These reports include diverse information on the request data, for example, throughput figures, such as the number of requests per day, week or month. The built-in reports use the data in the InstantLink database, and it depends on the cleaning interval how far back the report data covers.

In addition, InstantLink provides an option for InstantLink Archive that can be used for customer specific analysis and reporting. InstantLink Archive uses a separate database in which old request and task data can be stored for a longer period of time without affecting the performance of InstantLink. For more information on InstantLink Archive, see *InstantLink Archive Functional Description*.

4.3.6 System Modes

The InstantLink system has two modes:

- production mode
- configuration mode

In production mode, users can monitor the system but cannot modify the system configuration. This mode can be used to ensure that no one will accidentally change the system configuration when the system is under production use. In production mode, distributed components can be started up or shut down with sufficient privileges.

In configuration mode, the system configuration can be changed online, without having to shut down the system. Requests are processed also in configuration mode.

All new configurations become effective when the user refreshes the system configuration. The system modes are changed through the InstantLink user interface.

4.3.7 High Availability

InstantLink supports high availability, meaning that InstantLink documentation provides instructions on how to install and configure InstantLink on high availability platforms. InstantLink supports both active-passive and active-active high availability configuration. In an active-passive high availability cluster system, if one of the instances fails, for example, because of a server error, the high availability software informs the other server about the failure and gives a command for it to take control.

In an active-active high availability configuration, there are two or more InstantLink instances on different nodes. Each node is able to offer a full service on its own. When more than one node is running, a load balancer shares the work amongst all active components. If a node becomes unavailable because of a hardware failure, the remaining nodes are able to continue the service. A clustered configuration offers continuous availability.

For more information on high availability configuration, see *Provisioning and Activation High Availability Configuration Guide*.

4.3.8 Configuration Import and Export

InstantLink allows its configuration data to be imported and exported within the same InstantLink version. The user can export the configuration to an XML file, from which the configuration can be imported to a new InstantLink instance or stored in a version management system. The configuration import and export are done through the InstantLink user interface or via command line.

4.3.9 Environment Information

Environment information of InstantLink and the operating system can be generated and downloaded through the InstantLink user interface. The InstantLink specific information includes alarms, component logs, crontab configuration, UI mode, GRC parameters, installations, processes, Java version, network configuration, request and task information, and status.

The InstantLink and operating system specific information includes all processes, disk utilisation, environment variables, general information, memory, patches, processors, system log and top processes.

In a distributed InstantLink environment, the environment information report gathers information from all hosts.

For more information on generating and downloading environment information, see *InstantLink Online Help*.

4.3.10 Upload and Download

InstantLink provides a tool for uploading and downloading system information through the user interface. Users with proper privileges can upload system configuration files such as Network Model, Global Resource Configuration (GRC), user information, request locking and service module registration files. These system configuration files can also be downloaded. In addition, users with proper privileges can download reports and system environment information through the user interface to their workstation.

For more information on uploading and downloading system information, see *InstantLink Reference Manual* and *InstantLink Online Help*.

4.3.11 Test Tool

Test Tool provides easy sending of test requests to InstantLink and collecting log information on the basis of that.

InstantLink includes a testing capability that collects information on processed requests from the Business Service Tool request trace and from the following NE logs: MML, I/O and error logs.

Test Tool can be used for:

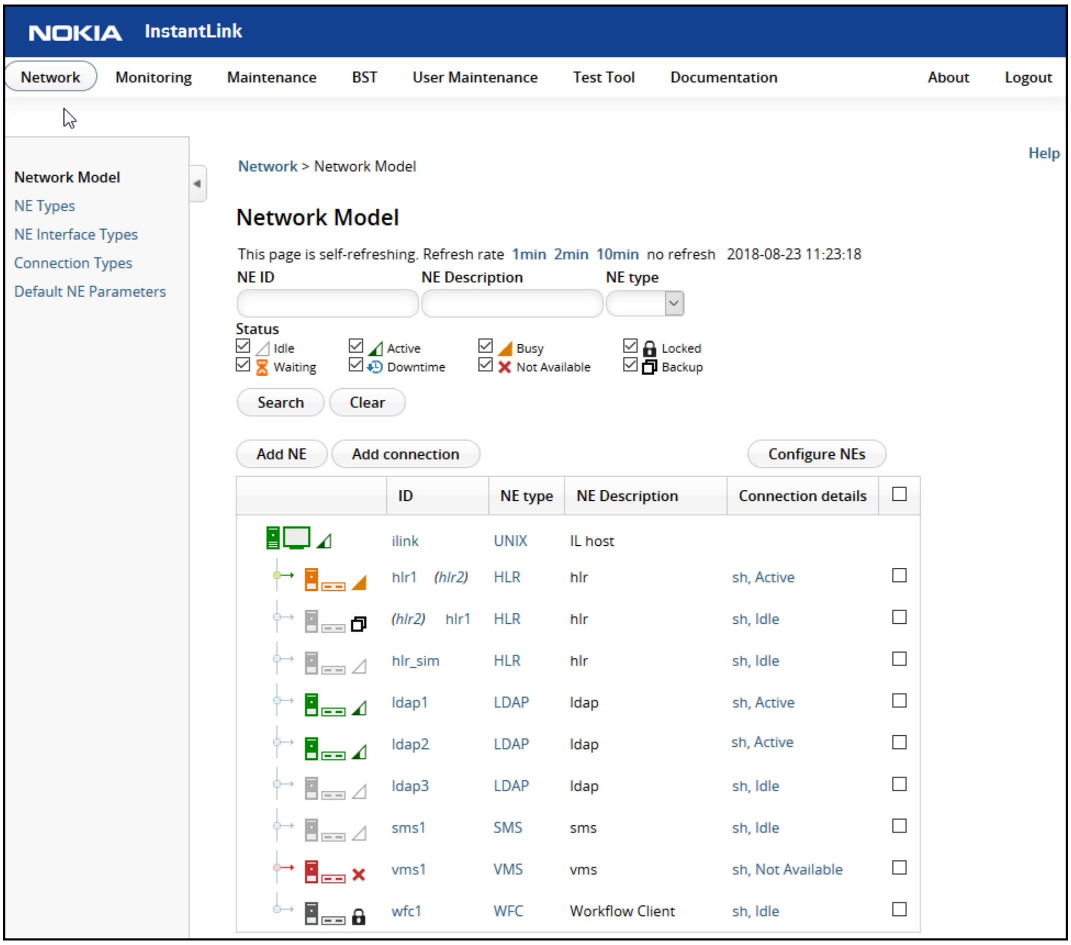
- defining and sending test requests
- viewing log files for test requests
- comparing test results

Test Tool is especially useful when configuring Business Service Tool logics but it must not be used in a production environment because it affects the performance of InstantLink. For more information on Test Tool, see *InstantLink Operation and Maintenance Guide*.

4.4 Network Model

To be able to send tasks to network elements, InstantLink needs to know about the underlying network. Network Model Manager (Nemo) is responsible for maintaining a network model. A network model consists of network elements and connections between InstantLink and network elements. It is used for representing a network topology of an operator's provisioned network. The network model is stored in the InstantLink database and it is read every time InstantLink starts up or system refresh is performed.

InstantLink user interface provides the network elements in a listed format on the **Network Model** page. *Figure 8* presents the **Network Model** page in the user interface.



Network Model

This page is self-refreshing. Refresh rate 1min 2min 10min no refresh 2018-08-23 11:23:18

ID	NE type	NE Description	Connection details	□
ilink	UNIX	IL host		
hlr1 *(hlr2)*	HLR	hlr	sh, Active	□
(hlr2) hlr1	HLR	hlr	sh, Idle	□
hlr_sim	HLR	hlr	sh, Idle	□
ldap1	LDAP	ldap	sh, Active	□
ldap2	LDAP	ldap	sh, Active	□
ldap3	LDAP	ldap	sh, Idle	□
sms1	SMS	sms	sh, Idle	□
vms1	VMS	vms	sh, Not Available	□
wfc1	WFC	Workflow Client	sh, Idle	□

Figure 8. Network Model

For more information on **Network Model**, see *InstantLink Online Help*.

4.4.1 Modifying Multiple Network Elements

A user with modify network settings privileges can modify multiple network elements at once in the InstantLink user interface. It is possible to lock, unlock and delete multiple network elements at the same time, and to change the time windows settings of several network elements. For more information, see *InstantLink Online Help*.

4.4.2 Time Windows

InstantLink provides a possibility to define a time window for each network element. The time window is used to identify if the network element may be accessed by InstantLink or not. The time window functionality can be used, for example, when a network element is taken out of service for a planned maintenance break.

With configurable time windows, operators can define when a specific network element is active (up-time window) or inactive (down-time window). When a network element is inactive, InstantLink does not send service requests to that network element but stores them and sends the stored service requests when the network element is active again. It is also possible to reroute tasks to a backup network element when a network element is inactive.

For more information on time windows, see *InstantLink Reference Manual*.

4.4.3 Backup Network Elements

In InstantLink, it is possible to configure one backup for each network element. If a network element, or the connection to it, is unavailable, the tasks queuing to that network element are transferred to the backup network element.

The network element and its backup form a pair of active and passive network elements. The active network element receives tasks from the queue, and the passive network element is waiting to turn active when needed. When the active network element becomes unavailable, the passive network element takes over and becomes active until manually locked or otherwise unavailable.

Backup network element functionality can be combined with time windows. During the downtime of a given network element, traffic is rerouted to its backup network elements.

For more information on backup network elements, see *InstantLink Reference Manual*.

4.4.4 Virtual Network Elements

InstantLink Network Model provides a possibility to define virtual network elements (VNEs). A VNE is a symbolic network element, which can formally represent several network elements of the same type in the network model. The address of a network element, for example IP address, is passed as a parameter in tasks, and VNEs ensure controlled task queuing and sending to network elements. Using VNEs decreases the size of the network model and makes the overall processing more efficient in case there are, for example, several hundred network elements of the same type. Typically, VNEs are used when InstantLink uses a network inventory to fetch information about the physical elements in the network.

Task Engine is responsible for receiving and forwarding tasks headed to VNEs. Task Engine first receives data from Network Model Manager and checks whether tasks are to be directed to VNEs. If Task Engine finds tasks going to a VNE, it sends them to the target VNE.

You can also build a Network Model consisting of managed network elements and VNEs. However, a single network element cannot act in both roles.

For more information on the virtual network elements, see *InstantLink Online Help*.

4.5 Network Element Interface

Network element interfaces (NEIs) are separately installable additional products. As the number of network element types grows, a new network element interface can be deployed quickly, because modifications in the InstantLink core system are not necessary.

InstantLink translates service provisioning tasks to network element specific messages (such as MML, SOAP, XML, Java RMI calls, CORBA IIOP or Q3 messages) in the NEI modules, and sends the messages to appropriate network elements.

InstantLink also interprets the responses received from the network elements and stores them in its database. InstantLink sends the response including the result of the task execution to OSS/BSS systems.

InstantLink supports several network elements and several element versions from different vendors simultaneously.

4.6 OSS/BSS Interface

The OSS/BSS interfaces are responsible for communication between OSS/BSS systems and InstantLink.

InstantLink provides a built-in interface, Extended API, for receiving service requests from OSS/BSS:

InstantLink also provides technology based APIs, such as InstantLink SOA Web Services and Batch API. In addition, InstantLink provides vendor-specific APIs for the industry's key CRM and order management systems. These APIs are optional products and have to be installed separately.

An OSS/BSS interface converts request messages into InstantLink internal format for further processing. Similarly, the OSS/BSS interface converts response messages from InstantLink internal format to the OSS/BSS format.

InstantLink provides a braking mechanism for request traffic between OSS/BSS systems and InstantLink. When clients are sending a large number of requests and InstantLink’s internal queues reach predefined thresholds, InstantLink slows down the sending of request acknowledgements back to the OSS/BSS systems in order to avoid congestion.

4.6.1 Provisioning Bandwidth to OSS/BSS Client Systems

InstantLink allows allocating provisioning bandwidths for client systems belonging to an organisation. All organisations get a defined proportion of throughput and, for instance, during peak hours of request processing, a certain proportion of the throughput is allocated to each client system used. Allocating provisioning bandwidths to OSS/BSS client systems is necessary when InstantLink is running in a multi-operator environment.

For more information on provisioning bandwidths, see *InstantLink Operation and Maintenance Guide*, *InstantLink Reference Manual* and *InstantLink Online Help.*

4.7 Asynchronous Network Element Interfaces

InstantLink can be configured to interface with asynchronous network elements. Asynchronous network element interface means that instead of immediately receiving a response from a network element for a provisioning command, the network element processes the command for some time (from minutes to days) and then sends a response through a dedicated return channel. InstantLink includes two different return channels:

- Response FIFO
- RMI Response FIFO

The Response FIFO allows receiving different asynchronous task responses simultaneously from one source.

The RMI Response FIFO enables receiving task responses from several different sources at the same time. The RMI FIFO includes a braking mechanism which ensures that the task queue does not exceed its limit which could slow down system performance. Using the RMI Response FIFO is an advantage for an operator that handles high volumes of asynchronous network element tasks. For more information on Response FIFO and RMI Response FIFO functionalities, see *InstantLink Reference Manual*.

4.8 Access Control and Security

InstantLink provides two types of user authentication:

- OSS/BSS system authentication for a system using a northbound client API
- UI user authorisation

The OSS/BSS user authentication determines the sending of requests. The user interface user authorisation means defining which pages in the user interface certain users are allowed to access.

4.8.1 OSS/BSS System Authentication

OSS/BSS systems are authenticated when they try to open a connection to InstantLink for sending requests. If the authentication fails, an error message is sent back to the OSS/BSS including the reason for the error, and the connection is closed.

In InstantLink, a response queue ID is defined for each client user. This enables specifying which response queue IDs client users are allowed to use when sending requests.

Client users that belong to an organisation can only use the response queue IDs that are defined for the organisation. For more information on organisations, see Section 4.1.9 *Supporting Virtual Network Operators and Multiple Organisations*.

4.8.2 User Interface Authentication

InstantLink user interface users can be authenticated either with the internal InstantLink user database or with an external authentication server. It is possible to use mixed mode meaning that part of the user interface users are authenticated with the InstantLink user database and part with an external authentication server. The Administrator user is always authenticated with the InstantLink user database.

4.8.2.1 Internal Authentication

InstantLink user authorisation and user authentication follow the typical Sarbanes-Oxley (SOX) requirements. User authorisation limits the user's possibilities to perform operations in InstantLink. User authentication is based on a user name and password. Each user password is stored into the system in a SHA-256 hashed form. New users must change their password when they log in to the system for the first time.

UI profiles define the privileges that each UI user has in the system. They define, for example, which users are allowed to modify the system configurations, manage the user profiles or monitor the system.

UI profiles are configured to use a password profile. Password profiles determine how secure passwords UI users must use. With password profiles, the administrative user can define the following:

- minimum length of passwords
- password must contain numbers

- password must contain special characters
- password must contain upper and lower case characters
- password must not contain the user ID
- passwords cannot be re-used. The Administrator user can define how many old passwords system must remember.
- define a period of time in days when the user will be notified about password expiration
- define the maximum number of failed login attempts before the user account is locked
- define the automatic deactivation interval of an inactive account in days
- allow only one concurrent user interface session per user

For more information on password profiles, see *InstantLink Online Help*.

If a user tries to log in to the system with a valid user ID but with an incorrect password, the user account is locked after a configurable number of attempts. This number is configured in the password profile. The administrator of the system must unlock the user account before it can be used again.

4.8.2.2 Direct External Authentication

When a user interface user is configured to use an external authentication server, all authentication rules are defined in the external authentication server.

The user account must be created both in the InstantLink user database and in the external authentication server. After this the password control and authentication is the responsibility of the external authentication. Users can change their password centrally in the external authentication server. Additionally, administrator can lock and unlock user accounts from the external authentication server.

For more information about configuring the use of external authentication servers, see *InstantLink Operation and Maintenance Guide* and *InstantLink Online Help*.

4.8.2.3 Brokered External Authentication

External authentication using OpenAM as an authentication broker provides InstantLink with the ability to communicate indirectly with heterogenous user management systems through a unified interface, and offloads the complexity of configuring access to those systems to OpenAM.

Unlike the direct form of external authentication, which requires that externally authenticated user accounts exist both in the directory and in InstantLink, the brokered form does not have this requirement. It does however require that a suitable user profile is created for each user group in the directory service to be used with external authentication. The user profile must be configured with the same name as the group associated with an external user in the directory, and is used to manage the set of permissions allocated to an externally authenticated user.

4.8.2.4 Single Sign-On

Single sign-on (SSO) functionality allows InstantLink to share an authenticated session with one or more additional components within an SSO realm. A user needs only log in once to any single component in the realm in order to gain access to all components, or not at all if an existing session is ongoing for that user and can be resumed. The SSO functionality is an extension to brokered external authentication using OpenAM, and can only be used in conjunction with this component.

4.8.3 Security

It is possible to configure the InstantLink web server to use a secured communication, HTTP Secure (HTTPS), between the server and browser. For more information, see *Provisioning and Activation Installation Guide* and *InstantLink Operation and Maintenance Guide*.

It is also possible to configure InstantLink to use a secured communication between InstantLink and the network elements, depending on the network element interface protocol.

InstantLink can use Transport Layer Security (TLS) for sending Extended API request messages when in connection with the client system interfaces. The TLS connection ensures that requests and tasks using Extended API protocol are processed over an encrypted connection. For more information on TLS/SSL connections, see *InstantLink Reference Manual* and *InstantLink Online Help*.

4.8.4 Audit Trail

InstantLink keeps an audit trail that can be viewed through the user interface. The audit trail stores the configuration audit trail, from which the user can see what configuration actions have been performed in the system, who has performed them and when. The audit trail also shows when a user has logged in to or out of the user interface.

4.9 Fault Tolerance

InstantLink is a fault tolerant system that can handle:

- network errors
- task resending
- alarm sending
- abrupt shutdown
- component crash

InstantLink can be configured to send alarms to Event Management. For more information on Event Management, see the Event Management documentation.

The handling of errors is described in more detail in the following sections.

4.9.1 Network Errors

If an error occurs in a network connection, InstantLink detects the error and tries to reconnect to the network element. If these attempts fail, InstantLink does not unnecessarily try to use the route until it detects that it is working again. For this purpose, InstantLink checks the route periodically.

Even if there is no connection to the network element, InstantLink can still receive requests and store them in the database. Once the network element connections have been restored, InstantLink continues the request processing.

If a network connection breaks while InstantLink is executing a task, the system stores the task's status in its database. Once the connection is working again, the system uses the task's status information to continue its execution from the point when the connection was lost.

InstantLink enables defining a backup network element for managed network elements. When an error occurs in the network element connections, all tasks are routed to the backup network element until the default network element is recovered.

It is possible to configure a task expiration time, which is the maximum time in which the system tries to start the execution of a task. If a network element is unavailable for a long time, the task for the element can exceed its expiration time and the task fails. For more information on the InstantLink network errors, see *InstantLink Reference Manual*.

4.9.2 Task Resending

InstantLink sends an error message back to the OSS/BSS if the requests or tasks sent to it are not correct or authentication failed. If only some tasks in a request fail, for example, due to incorrect parameters or a network element error, the request is partly failed and the failed tasks can be resent through the InstantLink user interface.

4.9.3 Abrupt Shutdown

Normally, if InstantLink is shut down while there are still tasks in execution, all tasks that are queuing to network elements are stored in the database. InstantLink handles an abrupt system shutdown so that when InstantLink is restarted, all started requests are retrieved from the database. Also interrupted execution processes recover after the shutdown.

4.9.4 Component Crash

If Request Processing Engine, Network Model, Task Engine or ActiveMQ crashes in a distributed environment, it will be automatically restarted. Restart happens immediately for all components except Service Module Engines.

If a distributed component (Service Module Engine or Task Engine) crashes, the remaining component(s) must reclaim the unfinished work from the crashed component.

4.10 Service Modules

InstantLink is an open and flexible system that allows configuring request handling with service modules. Service modules offer customisable request and task processing functionality for the InstantLink user. For an illustration of the architecture of InstantLink and service modules, see Section 3.2 *Service Module Engine*.

With service modules, the InstantLink user can define how the system handles requests. Service modules include:

- Business Service Tool - A request service module that offers advanced business service functionality. It is capable of creating complex operations based on simple service requests. Business Service Tool derives tasks from requests according to provisioning logics, that is, rules defined by the user.
- InstantLink Routing Service Module – A task service module that allows defining routing rules to determine a target network element.
- InstantLink Logic Testing Tool - A task service module that provides simulation of responses from network element interfaces that can be used to test provisioning logics.

Business Service Tool is a separate add-on module with its own installation package. InstantLink Routing Service Module and InstantLink Logic Testing Tool are included in the InstantLink installation.

These service modules are introduced in more detail in the following sections.

4.10.1 Business Service Tool

Business Service Tool processes requests from OSS/BSS systems and creates request responses to them. Business Service Tool enables complex request processing based on provisioning logics that can be administered via the web-based user interface.

Business Service Tool gives operators concrete tools for configuring their provisioning logics in InstantLink through a web-based user interface. The provisioning logic is a set of rules that define how provisioning is performed. In Business Service Tool, a provisioning logic is first constructed using the Business Service Tool user interface and then activated to the InstantLink system. In this way, the entire InstantLink request-handling logic can be easily controlled.

For more information on Business Service Tool, see the Business Service Tool documentation.

4.10.2 InstantLink Routing Service Module

InstantLink Routing Service Module is used for routing tasks to specified network elements. When the network evolves, the exact details can be defined in InstantLink. This means that there is no need to make changes to the OSS/BSS when new network elements are added to the network.

The InstantLink system uses routing rules to send a task to the specified network element if no target network element has been defined in the request. Routing rules are defined with number ranges and configured in InstantLink Routing Service Module. In InstantLink Routing Service Module, the operator can also configure a numbering plan so that new network elements can be easily and efficiently added to the network.

For more information on InstantLink Routing Service Module, see the InstantLink Routing Service Module documentation.

4.10.3 InstantLink Logic Testing Tool

InstantLink Logic Testing Tool is used for simulating network element interface responses, for example, when testing provisioning logics. InstantLink Logic Testing Tool enables testing without installing network element interfaces, and responses can be tested without interacting with the actual network. Using InstantLink Logic Testing Tool, it is easy to modify existing test responses and to configure new test responses.

InstantLink Logic Testing Tool offers two different modes in which to generate responses: **automatic** and **interactive** modes. In the automatic mode, a request is created based on preconfigured rules. In the interactive mode, task rules are configured online during request execution.

For more information on InstantLink Logic Testing Tool, see the InstantLink Logic Testing Tool documentation.