SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Internet of things and smart cities and communities – General

SERIES F: NON-TELEPHONE TELECOMMUNICATION SERVICES

# Machine socialization: Relation management models and descriptions

Recommendation  ITU-T  Y.4002/F.748.3

ITU-T Y-SERIES RECOMMENDATIONS

**GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES**

| | |
|---|---|
| GLOBAL INFORMATION INFRASTRUCTURE | |
| General | Y.100–Y.199 |
| Services, applications and middleware | Y.200–Y.299 |
| Network aspects | Y.300–Y.399 |
| Interfaces and protocols | Y.400–Y.499 |
| Numbering, addressing and naming | Y.500–Y.599 |
| Operation, administration and maintenance | Y.600–Y.699 |
| Security | Y.700–Y.799 |
| Performances | Y.800–Y.899 |
| INTERNET PROTOCOL ASPECTS | |
| General | Y.1000–Y.1099 |
| Services and applications | Y.1100–Y.1199 |
| Architecture, access, network capabilities and resource management | Y.1200–Y.1299 |
| Transport | Y.1300–Y.1399 |
| Interworking | Y.1400–Y.1499 |
| Quality of service and network performance | Y.1500–Y.1599 |
| Signalling | Y.1600–Y.1699 |
| Operation, administration and maintenance | Y.1700–Y.1799 |
| Charging | Y.1800–Y.1899 |
| IPTV over NGN | Y.1900–Y.1999 |
| NEXT GENERATION NETWORKS | |
| Frameworks and functional architecture models | Y.2000–Y.2099 |
| Quality of Service and performance | Y.2100–Y.2199 |
| Service aspects: Service capabilities and service architecture | Y.2200–Y.2249 |
| Service aspects: Interoperability of services and networks in NGN | Y.2250–Y.2299 |
| Enhancements to NGN | Y.2300–Y.2399 |
| Network management | Y.2400–Y.2499 |
| Network control architectures and protocols | Y.2500–Y.2599 |
| Packet-based Networks | Y.2600–Y.2699 |
| Security | Y.2700–Y.2799 |
| Generalized mobility | Y.2800–Y.2899 |
| Carrier grade open environment | Y.2900–Y.2999 |
| FUTURE NETWORKS | Y.3000–Y.3499 |
| CLOUD COMPUTING | Y.3500–Y.3999 |
| INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES | |
| **General** | **Y.4000–Y.4049** |
| Definitions and terminologies | Y.4050–Y.4099 |
| Requirements and use cases | Y.4100–Y.4249 |
| Infrastructure, connectivity and networks | Y.4250–Y.4399 |
| Frameworks, architectures and protocols | Y.4400–Y.4549 |
| Services, applications, computation and data processing | Y.4550–Y.4699 |
| Management, control and performance | Y.4700–Y.4799 |
| Identification and security | Y.4800–Y.4899 |

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

# Recommendation ITU-T Y.4002/F.748.3

## Machine socialization: Relation management models and descriptions

**Summary**

Recommendation ITU-T Y.4002/F.748.3 specifies the relation management models and descriptions for machine socialization which enables machines to cooperate to achieve a given task using their relations with other machines. This Recommendation also includes use cases of relation management models and presents schemas to describe a relation.

**History**

| Edition | Recommendation | Approval | Study Group | Unique ID* |
|---------|----------------|----------|-------------|------------|
| 1.0 | ITU-T Y.4002/F.748.3 | 2015-11-29 | 16 | 11.1002/1000/12622 |

**Keywords**

Machine socialization, relation management model.

---

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

FOREWORD

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

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

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

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

NOTE

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

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

INTELLECTUAL PROPERTY RIGHTS

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

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

**Table of Contents**

# Recommendation ITU-T Y.4002/F.748.3

## Machine socialization: Relation management models and descriptions

## 1 Scope

This Recommendation specifies the relation management models and descriptions for machine socialization which enables machines to cooperate to achieve a given task using their relations with other machines. This Recommendation covers the following:

– relation management models for machine socialization;

– relation descriptions for machine socialization; and

– use cases for relation management models.

## 2 References

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

[ITU-T Y.4001]     Recommendation ITU-T Y.4001/F.748.2 (2015), *Machine socialization: Overview and reference model*.

## 3 Definitions

### 3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

**3.1.1 Internet of things (IoT)** [b-ITU-T Y.4000]: A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.

NOTE 1 – Through the exploitation of identification, data capture, processing and communication capabilities, the IoT makes full use of things to offer services to all kinds of applications, whilst ensuring that security and privacy requirements are fulfilled.

NOTE 2 – From a broader perspective, the IoT can be perceived as a vision with technological and societal implications.

**3.1.2 machine socialization** [ITU-T Y.4001]: Enabling things or machines to communicate what they do or what they need to each other as well as to follow each other, discuss with each other and collaborate with each other.

**3.1.3 relation** [ITU-T Y.4001]: An association between or among machines or things enabling machines or things to share or to provide the capability to achieve a task in collaboration. This includes scheduling of processes between or among machines or things to perform a task.

### 3.2 Terms defined in this Recommendation

This Recommendation defines the following terms.

**3.2.1 capability parameter**: Information on a capability of a machine to achieve a given task such as an input parameter, an output parameter, processing time, etc.

**3.2.2    capability set**: A set of capabilities that machines can provide to process commands from users.

**3.2.3    machine profile schema**: A schema to describe features of a machine.

**3.2.4    relation module**: A module that acts as a relation server in the distributed relation management model. A relation module resides in every machine in the distributed relation management model.

**3.2.5    relation profile schema**: A schema to describe a relation to achieve a given task. It contains information on a group of machines, processing schedules, workgroup IDs, work descriptions and a capability set.

**3.2.6    relation server**: A server to establish a relation and operate machines according to the relations.

**3.2.7    status parameter**: The current status of a machine such as active, standby, expected termination time of the current operation, current execution task, current execution function and expected termination time of the current execution function.

# 4      Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

XML    Extensible Markup Language

# 5      Conventions

None.

# 6      Overview of relation for machine socialization

Like the human experience of using social network services to obtain information on the knowledge and capabilities of other people, unrestricted by barriers of time and place, etc., machines can communicate what they do or what they need; they can follow each other, discuss, collaborate, create events and do things together. This involves the socialization of machines to a level corresponding to that of the social relations of humans [ITU-T Y.4001].

Socialization may have many meanings, but one of the most important features of socialization is working together. Machine socialization implicates that machines work together to achieve given tasks. To achieve the task in collaboration, machines have to establish associations with other machines by sharing and providing capabilities, these associations are called relations in [ITU-T Y.4001] and in this Recommendation.

In other words, machines should exchange data with other machines and understand the meaning of the exchanged data to establish a relation.

Humans can exchange their ideas with other people using languages or gestures, however a machine cannot exchange information with other machines using natural languages. Therefore, a machine should exchange information and respond to other machines by analysing the information with other machines that is expressed in a mutually understandable form, such as extensible markup language (XML).

This Recommendation defines a schema for a mutually understandable form of machine socialization.

If machines can understand and exchange data with other machines, a relation can be established through relation management.

Relation management establishes a relation, operates machines using the established relation and releases the relation as specified by relation management models.

Procedures to establish a relation vary according to relation management models.

## 7 Relation management models

For machine socialization, it is necessary to manage relations among machines. Relation management varies depending on the relation models. Relation management models may require user intervention in cases where the results of tasks are critical to the user.

In this Recommendation, relation management models define creation and release of relations in three different ways based on the machine socialization framework defined in [ITU-T Y.4001].

### 7.1 Centralized relation management model



Y.4002-F.748.3(15)_F01
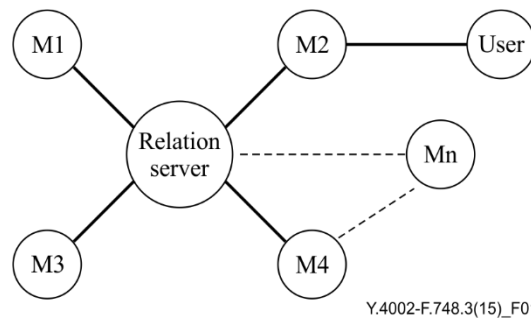
**Figure 1 – Centralized relation management model**

In the centralized relation management model shown in Figure 1, a relation server manages relations among machines and executes a task following the steps listed below:

1) Machines register machine profile parameters to the relation server.
2) A machine receives a user's command to execute a task and forwards the task to the relation server.
3) The relation server analyses the task.
4) The relation server establishes a capability set to execute the task.
5) The relation server establishes a group of useable machines to execute the task based on the capability set, capability parameters and status parameters of the machines that are already registered on the server or extractable from the target machines.
6) The relation server establishes a relation for the group based on the capability set, the capability parameters and the status parameters of the member machines.
7) The relation includes the group of machines and the schedules of the processes necessary to perform the task.
8) The relation server establishes a relation profile based on the relation.
9) If the results of the processes in the relation profile are critical to the user, the relation server should request a user intervention and update the relation profile:

   a) The relation server requests the user to approve the performance of the task or to select processes by listing processes executed by the machines.

   b) The user can either approve the performance of the task or select the processes to permit the performance of the task and then notifies the relation server.

   c) The relation server may establish a new group of machines and a new relation based on the user's decision.

d) The relation server establishes a new relation profile.

10) The relation server commands the machines to execute the processes following the schedule of processes included in the relation profile.

11) After completion of the task, the relation server releases the machines from the relation.

## 7.2 Distributed relation management model



Y.4002-F.748.3(15)_F02
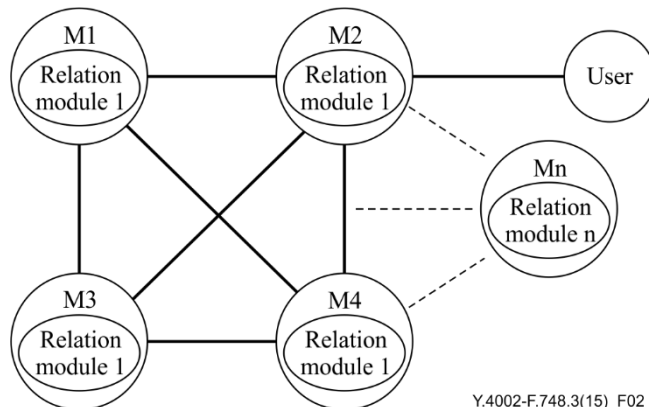
**Figure 2 – Distributed relation management model**

In the distributed relation management model shown in Figure 2, a relation is established by a relation module that resides in each individual machine. The machine which receives a user's command to perform a task forwards the task to other machines to create a relation. Each machine should determine whether to join the relation or not to execute the task. The machine that received the user's command acts as the coordination machine to create a relation based on the capability sets, capability parameters and status parameters of the group of machines in the relation.
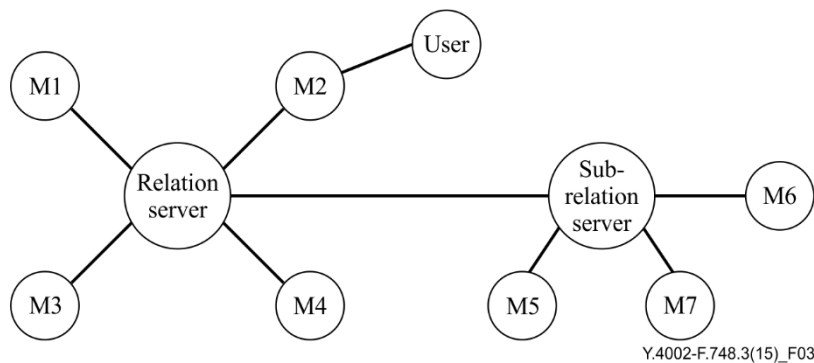
The relation module creates and manages the relation and executes a task following the steps listed below:

1) A machine receives a user's command to execute a task from the coordinating machine and forwards the task to the other machines.

2) The relation module in each individual machine analyses the task and makes its own capability set to execute the task.

3) Each relation module decides whether to join the group of machines to execute the task, based on its capability set, capability parameter and status parameter.

4) If a relation module decides to join the group, the relation module shall notify the coordination machine and send its capability set, capability parameters and status parameters to the coordinating machine.

5) The coordinating machine's relation module creates a relation based on the received capability sets, capability parameters and status parameters of the group of machines.

6) The relation includes the group of machines and the schedule of processes necessary to perform the task.

7) The coordinating relation module creates the relation profile and forwards the relation and the relation profile to the member machines.

8) If the results of processes in the relation profile are critical to the user, the machine that received the user's command should request a user intervention and update the relation profile:

a) The coordinating machine requests the user to approve performance of the task or to select processes by listing processes executed by the machines.

b)   The user can either approve performance of the task or select the processes and then notifies the coordinating machine.

c)   The coordinating machine notifies the user's decision to other machines.

d)   The relation modules in individual machines make new decisions to join the relation and forward necessary information to the coordinating machine.

e)   The coordinating relation module creates a new group of machines and a new relation based on the user's decision.

f)   The coordinating relation module establishes a new relation profile.

9)   Grouped machines should operate by following the schedule of processes included in the relation profile.

10)   After completion of its own task, each machine releases itself from the relation.

## 7.3   Nested-centralized relation management model

The nested-centralized relation management model extends the centralized relation management model.



**Figure 3 – Nested-centralized relation management model**

Figure 3 shows an example of the nested-centralized relation management model where seven machines and two relation servers are involved.

The model manages a relation and executes a task following the steps listed below:

1)   A machine registers machine profile parameters to either the relation server or to the sub-relation server.

2)   A machine receives a user's command to execute a task and forwards the task to the relation server.

3)   The relation server forwards the task to the sub-relation server.

4)   The relation server and the sub-relation server analyse the task.

5)   The relation server and the sub-relation server establish a capability set to execute the task.

6)   The relation server and the sub-relation server create a group of useable machines based on the capability set, capability parameters and status parameters of the member machines that are already registered on the server or extractable from the target machines.

7)   The sub-relation server establishes a relation of a group of machines and forwards it to the relation server.

8)   The relation server establishes a relation using its group of machines and the relation received from the sub-relation server.

9)   The relation includes information of the group of machines and the schedule of processes.

10) The relation server establishes a relation profile according to the relation.

11) If the results of processes in the relation profile are critical to the user, the relation server should request a user intervention and update the relation profile:

   a) The relation server requests the user to approve performance of the task or to select processes by listing processes executed by the machines.

   b) The user can either approve the performance of the task or select the processes and then notifies the machine.

   c) The relation server may establish a new group of machines and a new relation based on the user's decision.

   d) The relation server establishes a new relation profile.

12) The relation server forwards the relation profile to the sub-relation server.

13) The relation server and the sub-relation server command a group of machines to execute processes following the schedule of processes included in the relation profile.

14) After the completion of the task, the relation server and the sub-relation server release the machines from the relation.

## 8 Relation descriptions

The machine profile schema enables the relation server or relation module to mutually understand the features of machines to establish a relation and the relation profile schema stores the relation established.

## 8.1 Machine profile schema

The machine profile schema is a template to describe machines in terms of status, capabilities, IDs, interfaces, etc., which are given in Table 1. A relation is established by a relation server or a relation module with the information inscribed using this profile.

**Table 1 – Machine profile schema**

| Machine profile parameter | Sub-parameter |
|---|---|
| Status | Active |
| | Standby |
| | Current process |
| | Expected termination time of the current process |
| | Current function |
| | Expected termination time of the current execution function |
| Capability | Name of process |
| | Input parameter |
| | Output parameter |
| | Processing time (duration) |
| | Processing condition |
| | Function 1 |
| | Function 2 |
| | ……. |
| | Function n |
| | End of capability |

**Table 1 – Machine profile schema**

| Machine profile parameter | Sub-parameter |
|---|---|
| Machine ID | |
| User ID | |
| Group ID | |
| Operating system | |
| Machine interface | Interface protocol between machines |
| | Interface parameter between machines |
| | Interface protocol between machine and relation server |
| | Interface parameter between machine and relation server |
| End of machine profile parameter | |

Machine profile parameters are described below:

– Status: Status of a machine indicating active, standby, the expected termination time of the current process, the current execution process, the current execution function and the expected termination time of the current execution function.

– Capability: Capability of a machine indicating the name of the task, input parameter, output parameter, processing time, processing condition and functions.

– Machine ID: Alphanumeric identifier of a machine which distinguishes a machine from another machine.

– Group ID: Alphanumeric identifier of a group of machines in a relation.

– Operating system: Operating system running on a machine.

– Machine interface: Communication protocols that a machine can support.

– End of machine profile parameter: Indicator to specify the end of the machine profile.

## 8.2 Relation profile schema

The relation profile schema shown in Table 2, is a template to describe a relation in terms of capability set, grouped machines, workgroup ID, task description and task processing schedule, etc. After a relation is established, relation information is stored in a relation profile. Tasks to be executed by a relation in machine socialization are expressed in a relation profile and are achieved according to the schedule of processes defined in the relation.

**Table 2 – Relation profile schema**

| Relation profile parameter | Sub-parameter |
|---|---|
| Capability set | Capability 1 |
| | … |
| | Capability n |
| | End of capability set |
| Grouped machines | Machine ID 1 |
| | …. |
| | Machine ID n |
| | End of machine ID |

**Table 2 – Relation profile schema**

| Relation profile parameter | Sub-parameter |
|---|---|
| Workgroup ID | |
| Task description | |
| Task processing schedule | Process 1 |
| | Process start time |
| | Process start condition |
| | Allotted machine ID |
| | Start time of function 1 |
| | Functions 1 |
| | …. |
| | Start time of function n |
| | Function n |
| | End of functions |
| | Expected time of process termination |
| | Interface parameters |
| | Termination condition |
| | End of process 1 |
| | Process 2 |
| | Process start time |
| | Process start condition |
| | Allotted machine ID |
| | Start time of function 1 |
| | Functions 1 |
| | …. |
| | Start time of function n |
| | Function n |
| | End of functions |
| | Expected time of process termination |
| | Interface parameters |
| | Ending condition |
| | End of process 2 |
| | ….. |
| | Process n |
| | Process start time |
| | Process start condition |
| | Allotted machine ID |
| | Start time of function 1 |
| | Functions 1 |
| | …. |
| | Start time of function n |

**Table 2 – Relation profile schema**

| Relation profile parameter | Sub-parameter |
|---|---|
| | Function n |
| | End of functions |
| | Expected time of process termination |
| | Interface parameters |
| | Termination condition |
| | End of process n |
| | End of task processing schedule |
| End of relation profile parameter | |

Relation profile parameters are described below:

–     Capability set: The set of capabilities needed to complete the user's command to perform a task

–     Grouped machines: The group of machines which support the capability set.

–     Workgroup ID: Alphanumeric identifier of the workgroup.

–     Task description: Human readable description of the task implying a user's command.

–     Task processing schedule: Sequence of processes to complete the task.

–     End of relation profile parameter: Indicator to specify the end of relation profile parameter.

# Appendix I

## Use cases of relation management models

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

Three use cases of relation management models are introduced.

### I.1 Booking of a movie ticket based on a centralized relation management model

The use case shown in Figure I.1 illustrates the booking of a movie ticket using a centralized relation management model of machine socialization.



Y.4002-F.748.3(15)_FI.1
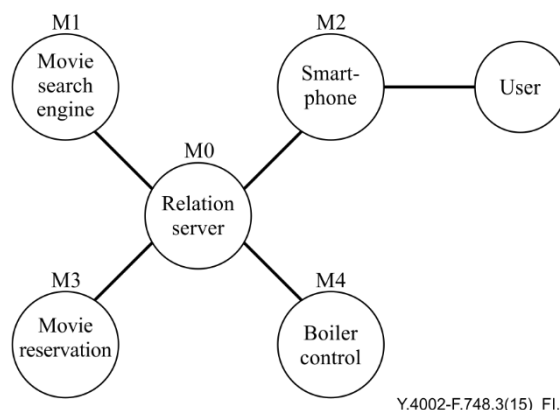
**Figure I.1 – Centralized relation management model**

Each machine can perform the following processes:

– M0: M0 manages relations among machines and executes a task.

– M1: M1 searches movies.

– M2: M2 receives the user's command to perform a task, booking a movie ticket. It forwards the task to the relation server.

– M3: M3 books movie tickets.

– M4: M4 controls the temperature of the house.

A relation is established among M0, M1, M2, M3 and M4 according to the following procedures:

1) M1, M3 and M4 have registered their machine profiles including searching capability, booking capability and controlling temperature capability to the relation server respectively.

2) M2 receives a user's command to perform a task and forwards the task to M0.

3) M0 analyses the task and determines a capability set which requires searching for a movie, booking the movie and controlling temperature. Controlling temperature capability will be used to ensure a comfortable environment for a user after watching a movie.

4) M0 creates a group of M1, M3 and M4 including M2 which forwarded the user's commands to M0.

5) With the assumption that the results of this task are not critical, M0 establishes a relation based on the capability set, the capability and the status in machine profiles of M1, M3 and M4 according to the following schedule.

6) Activation of M0, M1, M2, M3 and M4 by using the relation is as follows:

   a) M0 sends a command to M1 to search for the specified movie;

   b) M1 returns the result to M0 and M0 forwards the result to M2;

c)   The user selects a movie and the selection is forwarded to M3;

d)   M3 books a movie ticket with the confirmation of the user;

e)   M0 activates M4 at the time according to the schedule of the process;

f)   The relation is released.

## I.2    Booking of a movie ticket based on a distributed relation management model

The use case shown in Figure I.2 illustrates the booking of a movie ticket based on a distributed relation management model for machine socialization.
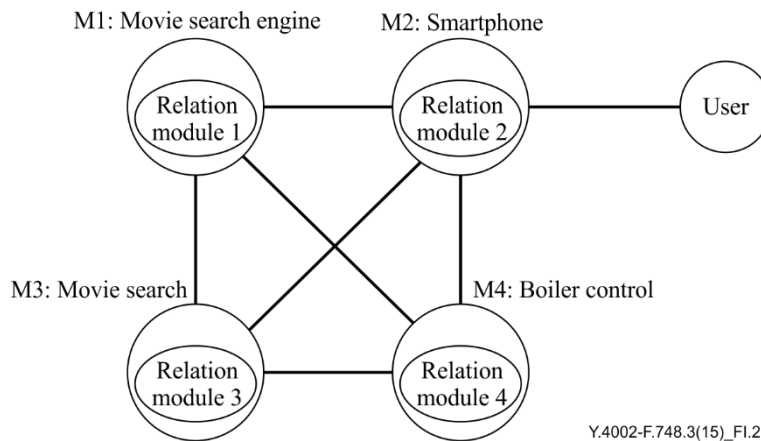


**Figure I.2 – Distributed relation model**

Each machine can perform the following processes:

–   M1: M1 searches for a movie.

–   M2: M2 is a coordinating machine and receives the user's command to perform a task, booking a movie ticket. It forwards the task to the other machines.

–   M3: M3 books movie tickets.

–   M4: M4 controls the temperature of the house.

A relation is established among M0, M1, M2, M3 and M4 according to the following procedure:

1)   M2 receives a user's command to perform a task and forwards the task to the M1, M3 and M4.

2)   The relation modules in the individual machines analyse the task and establish a capability set to execute the task.

3)   The relation modules decide whether to join the group of machines for executing the task coming from the user based on the capability set and its own machine capability parameter and machine status parameter.

4)   If the relation module decides to join the relation, it notifies M2 and forwards its capability set, the capability parameters and the status parameters.

5)   M2 creates a group of machines, a relation and a relation profile.

6)   The relation includes a group of machines and the schedule of processes to perform the task.

7)   With the assumption that the results of this task are not critical, the created relation and the relation profile are forwarded to M1, M3 and M4.

Activation of M1, M2, M3 and M4 by using the relation is as follows:

a)   M1 searches for the specified movie according to the relation profile;

b)	M1 returns the result to M2 and M2 forwards the result to the user;

c)	The user selects a movie and the selection is forwarded to M3;

d)	M3 books a movie ticket with the confirmation of the user;

e)	Relation module 4 in M4 activates M4 at the time according to the schedule of processes;

f)	The relation is released.

### I.3	Booking of a movie ticket based on a nested-centralized relation management model

The use case shown in Figure I.3 illustrates the booking of a movie ticket based on the nested-centralized relation management model for machine socialization.



Y.4002-F.748.3(15)_FI.3
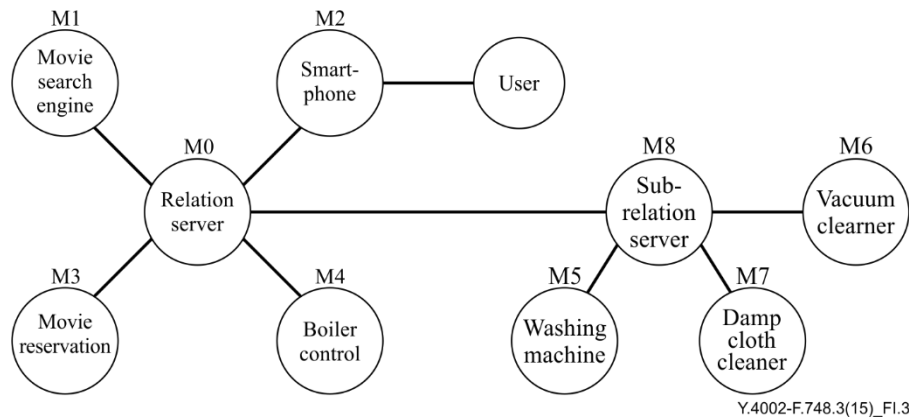
**Figure I.3 – Nested-centralized relation management model**

Each machine can perform the following processes:

–	M0: M0 manages relations among machines and executes a task.

–	M1: M1 searches for movies.

–	M2: M2 receives the user's command to perform a task, booking a movie. It forwards the task to the relation server.

–	M3: M3 books movie tickets.

–	M4: M4 controls the temperature of the house.

–	M5: M5 washes clothes.

–	M6: M6 performs vacuum cleaning.

–	M7: M7 performs damp cloth cleaning.

–	M8: M8 manages relations among submachines and executes a subtask.

A relation is established among M0, M1, M2, M3, M4, M5, M6, M7 and M8 according to the following procedure:

1)	M1, M2, M3, M4, M5, M6, M7 and M8 respectively have registered their machine profiles including searching capability, booking capability and controlling temperature capability to M0.

2)	M2 receives the user's command to perform a task and forwards the task to M0 and M8.

3)	M8 server analyses the received task and establishes a relation.

4)	M0 analyses the received task and grouping machines which can be used to execute the command by using machine profiles and the generated capability set.

5) With the assumption that the results of this task are not critical, a new relation is created in M0 based on the capability set, the capability, the status in other machine profiles and a relation which is generated by a sub-relation server.

6) M0 forwards the new relation profile to M8 server.

7) Activation of M0, M1, M2, M3, M4, M5 and M8 by using the new relation is as follows:

    a) M0 sends a command to M1 to search for the specified movie;

    b) M1 returns the result to M0 and M0 forwards the result to M2;

    c) The user selects a movie and the selection is forwarded to M3;

    d) M3 books a movie ticket with the confirmation of the user;

    e) M8 activates M5 at the time according to the schedule of the process;

    f) The relation is released.

# Bibliography

[b-ITU-T Y.4000]   Recommendation ITU-T Y.4000/Y.2060 (2012), *Overview of the Internet of Things*.

ITU-T F-SERIES RECOMMENDATIONS

**NON-TELEPHONE TELECOMMUNICATION SERVICES**

| | |
|---|---|
| TELEGRAPH SERVICE | |
| Operating methods for the international public telegram service | F.1–F.19 |
| The gentex network | F.20–F.29 |
| Message switching | F.30–F.39 |
| The international telemessage service | F.40–F.58 |
| The international telex service | F.59–F.89 |
| Statistics and publications on international telegraph services | F.90–F.99 |
| Scheduled and leased communication services | F.100–F.104 |
| Phototelegraph service | F.105–F.109 |
| MOBILE SERVICE | |
| Mobile services and multidestination satellite services | F.110–F.159 |
| TELEMATIC SERVICES | |
| Public facsimile service | F.160–F.199 |
| Teletex service | F.200–F.299 |
| Videotex service | F.300–F.349 |
| General provisions for telematic services | F.350–F.399 |
| MESSAGE HANDLING SERVICES | F.400–F.499 |
| DIRECTORY SERVICES | F.500–F.549 |
| DOCUMENT COMMUNICATION | |
| Document communication | F.550–F.579 |
| Programming communication interfaces | F.580–F.599 |
| DATA TRANSMISSION SERVICES | F.600–F.699 |
| **AUDIOVISUAL SERVICES** | **F.700–F.799** |
| ISDN SERVICES | F.800–F.849 |
| UNIVERSAL PERSONAL TELECOMMUNICATION | F.850–F.899 |
| HUMAN FACTORS | F.900–F.999 |

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

# SERIES OF ITU-T RECOMMENDATIONS

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