D 5.3: Description of Final UCN System Architecture and Proof-of-Concept Demonstrators

Due date of deliverable: 30.09.2016
Actual submission date: 30.09.2016

Start date of project: October 1st 2013
Duration: 36 months

Project Manager: Henk Heijnen, TRDF

Author(s): Fraunhofer FOKUS, University of Cambridge, University of Nottingham, Portugal Telecom, Technicolor, NICTA, Intamac, EURECOM, Martel Consulting

Editor: Fraunhofer FOKUS
Revision: v.1.0

Abstract

This deliverable describes the functional components developed in UCN and their relationship in the context of the system architecture. It also describes the showcase demonstrators developed in the project.

<table>
<thead>
<tr>
<th>Dissemination Level</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PU</td>
<td>Public</td>
</tr>
<tr>
<td>PP</td>
<td>Restricted to other programme participants (including the Commission Services)</td>
</tr>
<tr>
<td>RE</td>
<td>Restricted to a group specified by the consortium (including the Commission Services)</td>
</tr>
<tr>
<td>CO</td>
<td>Confidential, only for members of the consortium (including the Commission Services)</td>
</tr>
</tbody>
</table>

Project co-funded by the European Commission in the 7th Framework Programme (2007-2013)
# Table of Contents

1 Introduction .................................................................................................................. 1

2 Architecture Overview .................................................................................................. 2

3 Architecture Element Details ....................................................................................... 4

   Data Collectors ............................................................................................................. 4

   Personal Information Hub (PIH) .................................................................................. 5

      PIH Catalogue ........................................................................................................... 5

      PIH Store .................................................................................................................. 5

      PIH Gateway ............................................................................................................ 6

      Further services running on the PIH device ............................................................ 6

      PIH on the Cloud ....................................................................................................... 6

   Service Enablers .......................................................................................................... 7

      S&P Service Enablers ............................................................................................... 7

      AL Service Enablers ................................................................................................. 7

      TV Service Enablers .................................................................................................. 8

      User and Network Service Enablers ........................................................................ 8

4 Technical Changes since Year 2 .................................................................................... 8

   SmartData (PIH on the Cloud) .................................................................................... 8

4 Proof-of-Concept Demonstrators ................................................................................ 9

   Scenario 1: Population estimation .............................................................................. 9

      Description ............................................................................................................... 10

      Demonstration showcase .......................................................................................... 13

      Differences to Y2 Demo ............................................................................................ 13

      Placement in Architecture ......................................................................................... 14

   Scenario 2: Smart TV Movie Recommendation .......................................................... 15

      Description ............................................................................................................... 15

      Implementation ......................................................................................................... 16

      Differences to Y2 Demo ............................................................................................ 18

      Placement in Architecture ......................................................................................... 18

   Scenario 3: Assisted Living Services .......................................................................... 20

      Description ............................................................................................................... 20

      Implementation ......................................................................................................... 21

      Differences to Y2 Demo ............................................................................................ 22

      Placement in Architecture ......................................................................................... 22

   Scenario 4: Home network monitoring and diagnosis ................................................ 24

      Description ............................................................................................................... 24

      Implementation ......................................................................................................... 24

      Differences to Y2 Demo ............................................................................................ 26

      Placement in Architecture ......................................................................................... 26

5 Summary ....................................................................................................................... 28
1. INTRODUCTION

This document provides an update on the “D5.2 Preliminary System Architecture Description” deliverable and describes the status of the system architecture at the end of the project.

Initially, D5.2 was only intended to cover the system architecture itself, while this D5.3 document, titled “Description of Final UCN System Architecture and Proof-of-Concept Demonstrators” would add the description of the demonstrators. In addition to what was originally planned in the DoW, the project created an addendum to D5.2, describing the demonstrators to be implemented.

As a result, this D5.3 deliverable is in its whole an update of the D5.2 deliverable as delivered.

Structurally, the document consists of two parts - the system architecture and description of the proof-of-concept demos developed in UCN. Both parts are linked by architecture overview graphics, which highlight in which ways the individual demonstrators make use of the architecture to achieve their functionality.

The architecture links the four primary components developed in the UCN project.

- Personal Information Hub (PIH)
  This is the UCN core component, developed in WP1
- Data collectors
  These gather the data provided directly or indirectly by the user and were handled in WP2
- Service enablers
  These serve as mediators between services/applications and the PIHs of the individual users, avoiding the need for services/applications to contact the PIHs individually, covered in WP3
- Security and Privacy
- Services
  These are the applications built upon the UCN architecture, which are represented by the UCN proof-of-concept demonstrators, developed in WP5

This section is followed by the description of the four implemented demonstrators.

- Population estimation
  Showcasing the use of arithmetic on encrypted data
- Smart TV Movie Recommendation
  A system that can work with different levels of knowledge about the user
- Assisted Living Services
  TV based service that aids elderly and diseased users in monitoring their well-being
- Home network monitoring and diagnosis
  Assisting users and their ISPs in diagnosing network performance problems
2. ARCHITECTURE OVERVIEW

The primary units that comprise the UCN architecture are the data collectors, the Personal Information Hub and the cloud-based service enablers.

**Data collectors:** Depending on the specific use case, the UCN system employs different data collectors. Given that our goal is to gain contextual knowledge about the users, the data collectors must be running close to users. We are instrumenting user devices (both mobile and desktop/laptops) as well as information from the user’s home (through sensors and monitoring of the set-top box and the home gateway). In addition to information gathered at the user’s site, we also collect service specific information on the service backend (for example, users’ ratings of different movies). Ideally, all data generated by a user or a user’s environment would be captured by UCN data collectors so that we can get a more complete picture of the user context. In practice, however, different users may select to deploy just a subset of the data collectors. Data collectors send the stream of data securely to the PIH.

The light green box shows the data collectors. The primary data sources used within the home are TV, in-home sensor and home network based. Other data sources, such as social media activities or stated user preferences can provide further information to data collectors.

**Personal information hub:** The central element in the UCN architecture is the Personal Information Hub (PIH), a central repository of all data generated by the user. The PIH is a virtual platform, composed of multiple interacting components, which coordinate to provide the user with means to control access to their personal data by third parties (context consumers). User data may be stored on the PIH (in encrypted or unencrypted form), or stored...
encrypted on the cloud. The PIH implements a number of algorithms to give selective access to user data according to the user’s privacy settings. The PIH acts as an information broker between the user context data and the context consumers.

In the architecture image above, the Personal Information Hub is shown in the purple box. In the UCN project, the PIH is currently running on a physical device, a CubieTruck board utilizing MirageOS. The CubieTruck single-board computer can also run additional services if needed by applications. The unikernel concept implemented by MirageOS allows the provision of such additional services on the same physical device, without compromising the security of the PIH core function. Conceptually, the PIH functionality can also be provided by an existing hub device or a cloud service.

**Service enablers:** In most real-life scenarios, applications are unlikely to communicate with individual PIHs directly, so a number of services enablers will be handle the interfacing with multiple PIH instances and perform aggregation and recommendation services for applications. These service enablers are shown in the dark green box.

**Services and applications:** While not part of the architecture itself, third-party applications and services are the consumers of the rich user data stored in the PIHs, aggregated and provided by service enablers. These services and application deploy, for example, mechanisms for context-aware recommendations, and mechanisms for content delivery optimizations and service personalization.
3. ARCHITECTURE ELEMENT DETAILS

DATA COLLECTORS

UCN utilizes these data sources within the home environment.

IPTV Box
Provides data about viewing behaviour, based on user activity logs.

IPTV Mobile
Can provide same data as IPTV Box above, plus location information (current and historic).

EPG
Supplies metadata in combination with IPTV user activity logs and content rating.

User Preferences
Gathered by either the IPTV box or mobile, provides ratings the user gives the viewed content.

Social Networks
Provides data from user chatting with other users watching the same channel

Sensors
Information from a multitude of sensors in the home to gather contextual data about household routine(s), person presence and household environment.

Hostview
HostView is a data collection service running on end user devices, primarily allowing the gathering of network information from the terminal side.

Gateway OpenWRT
OpenWRT is extensible GNU/Linux distribution for embedded devices (typically wireless routers), allowing the gathering of network information on the router.

Gateway Monitoring
This software, running on an OpenWRT equipped router, performs capture and initial evaluation of network and connection information on the router.

Home collector
Represents any data collection device that is independent of the PIH or the router. Typically this will be a legacy home automation hub that aggregates data from a specific set of sensors and devices deployed in a home and combines data gathering and device control functionality.
PERSONAL INFORMATION HUB (PIH)

The purpose of the PIH is to provide users with means to manage access to personal data so that they can engage in the use of their personal data and so that their personal data can be better protected while still enabling the vast range of applications that wish to make use of personal data. As a federated system, the PIH is expected to run both on the cloud and local, physical devices.

The PIH functionality is supplied by three primary building blocks; the catalogue, the store and the gateway.

PIH CATALOGUE

The **PIH catalogue** enables users to publish the data sources they support while maintaining privacy over the existence of those data sources: only potential consumers of their data are able to discover the existence of a data source for a particular user.

For example, a user might publish the following record:

```
contact-email:unlinkable@example.com
gateway-url:https://data-sources.example.com/
data-source:contacts-v1
data-source:calendar-v2
data-source:location-v3
data-source:mediaprefs-v1
```

When requested for movie recommendations by a user (identified by email address), the recommendation service interested in the media preferences of that user can request access to `data-source:mediaprefs-v1`. The catalogue executes a search and, returns the associated gateway URL and an access token. The recommendation service can then use information to contact the user’s PIH Gateway to request access to the specified personal data.

PIH STORE

The **PIH Store** is a database on the PIH wherein a user’s personal data, coming from the data collection sources is stored. As this data represents collected information about the personal data of a user and allows detailed insights into a user’s behaviour, it is important that this storage is secure and will not be breached. In the current architecture, Irmin is currently used for data storage, since it is based on git-like concepts and inherently maintains a complete history of updates and accesses.

UCN implements an individual Irmin repository for every data source and runs each instance in its own unikernel, reducing the risk of exploiting database bugs to gain access to data beyond the permitted sources. In essence, every data source has its own individual, independent database. In addition, the Irmin database itself does not provide network connectivity. External connections to the Irmin data-store unikernels are relayed through MirageOS unikernels that effectively act as “application specific firewalls”.


**PIH GATEWAY**

The **PIH Gateway** is the element from which a data consuming service requests access to particular data, and by which a user grants access. Having advertised the availability of this data through the PIH Catalogue, the PIH Gateway responds to requests for access to that data. When a request to access data arrives at the PIH, it is handled by a gateway unikernel, which verifies its validity and requests permission from the user to process the request. If permission is denied, the request is refused. If permission is granted, the gateway device boots a request-specific firewall unikernel that bridges the datastore unikernel to the external network but ensures that only the specific consumer is able to connect and thus access the datastore, for the specific time period, after access has been specifically validated.

**FURTHER SERVICES RUNNING ON THE PIH DEVICE**

In environments where the PIH is represented by a physical device within the user’s home (within the scope of the project, the physical embodiment of the PIH is a CubieTruck running MirageOS), it is an ideal central hub and ‘go to point’ for UCN related applications. Since the concept of application specific unikernels ensures a secure and shielded environment for every application running on the hub, it is a useful device for running small applications, provided they don’t have extensive CPU or storage usage requirements.

Currently, three additional services run on the PIH, which, technically, are not part of the PIH itself.

- **MUSE** – support for multi-user searchable encryption, allowing searches over encrypted databases.

- **SPACE** - Sticky Policies Access Control and Encryption Scheme to allow encryption of data together with access control policies, allowing only consumers who fulfil the policies requirements to decrypt the data partially or totally.

- **Privacy Preserving Aggregator** – provides numeric information in an encrypted form that allows external data consumers to determine the aggregated sum of data from individual consumers without providing information about the individual values and their specific sources.

**PIH ON THE CLOUD**

The PIH concept can also be implemented on the cloud instead of at the users’ premises. In this case the user will have to trust the cloud provider but it is a solution for users who don’t want to buy/have a PIH at home, or in spite of having it choose to allow some of their data to be transferred to the cloud provider to profit from additional added value services as data aggregation and data analytics as well as more sophisticated recommendations and suggestions which may be offered by the cloud provider by cross checking data from different sources and users. As described in D5.2, Altice Labs (former PTIN) developed a secure platform on the cloud, called SmartData which sits in the middle of the data collection layer and the applications and implements additional functionalities as a context broker, storage
(also a big data infrastructure) and all the security mechanisms developed in the UCN project. This is an implementation of the PIH on the cloud which can also be seen as an extension of the PIH at the user’s premises. By being on the cloud and integrated with some Service Enablers (e.g. recommendations), SmartData aggregates information about all the users that choose to use this platform. This makes the information retrieval by services more agile and enriches the overall UCN services offer to their clients.

**SERVICE ENABLERS**

Most applications will not communicate directly with individual PIHs, but will make use of services enablers, which will be handle the interfacing with the PIH instances and perform aggregation and recommendation services for applications.

All service enablers utilize a context broker and ID management module for access to the PIH gateways.

The service enablers within UCN currently belong to four groups.

**S&P SERVICE ENABLERS**

This covers the enablers related to search and policies. They provide cloud based APIs for applications requiring search functionality across encrypted databases on multiple PIHs and handle access to data with sticky policies on those PIHs.

These enablers are, for example, used in the Assisted Living Services demonstrator, where care takers require access to sensitive medical data on the PIH and it is required (and often legally mandatory) that the data is only stored in encrypted form and only safely shared exclusively with the people who have explicit access permission by the user (e.g. doctors, caretakers, family).

**AL SERVICE ENABLERS**

This enabler provides a notification and advice service, based on health related sensor information from users. It primarily acts as an ‘intelligent filter’ between the health data available from a user and alarm services and services providing the user with health advice for specific situations (like exercises to do). In many cases, individual health data values may not be a cause for alarm by themselves, but can only be properly evaluated in the context of other health sensor historic data values. Inversely, health data values may be a cause for concern if they show a constant change for the worse, even though they might be within a ‘normal’ range.

The AL Recommend/Notify Service Enabler allows the specification of rules, which provide a threshold before external applications (like the Well Being Application) get notified.
TV SERVICE ENABLERS
The service enablers for TV applications currently consist of a recommendation and profile manager and a structured recommendation service, which together serve as a middleware platform where authorized services, such as recommendation and notification services, can subscribe and be notified of context changes in real-time and also have access to historical information if needed. Access control policies “filter” the kind of information available for each service/consumer - e.g. the TV movie recommendation service will only have access to the TV consumption user data. With access to movie ratings and mood, the system is able to filter similar movies based on user preferences, using Matrix Factorization.

USER AND NETWORK SERVICE ENABLERS
To aid the provision of information about potential network problems and the location of possible bottlenecks, two service enablers are provided. One of them is dedicated to processing the raw network data from the user’s environments to create meaningful metrics that enable an objective assessment of application quality on the end user’s devices. While this function can, conceptually, also be performed within the user’s network, it is computation and storage intensive, so in many cases the user will opt to have this performed by an external, cloud based service. A second enabler monitors performance on the service provider network to enable the Home Network Diagnosis software to determine whether perceived problems reside in the user’s home network or are due to outside factors.

TECHNICAL CHANGES SINCE YEAR 2
SMARTDATA (PIH ON THE CLOUD)
Due to the need of interoperability with more sensors, and use cases, the following components were modified since the last delivery:

- **Context Broker** - The message protocols supported by SmartData are: MQTT, AMQP, HTTP (REST), CoAP and now SOAP. There are different types of Providers: Proactive, Reactive and “on Demand”. The proactive providers are designed for devices that automatically send messages to the platform, whilst the reactives only respond upon request; so the platform will be in charge of constantly sending requests. The “on demand” is used when there is no need to frequently ask for updates but instead only on request by service enablers; it also allows to specify where to fetch information from and parameterize the requests if needed. Furthermore, in case the user wants to restrict publishing only to authorised external data providers, he can force them to authenticate themselves before start publishing data.

- **SEs (Service Enablers)** – The API now allows the search and simple manipulation of all JSON messages stored in the platform, in a SQL manner. It allows the filtering that correspond to SELECT, WHERE and ORDER, and the use of the operations that correspond to SUM, AVERAGE, MAX, MIN and COUNT.
• **Driver/NA (Network Adapter)** – With the addition of a search functionality in JSON data, the driver now has the possibility to convert non-JSON messages to a JSON format.

![SmartData Architecture (PIH on the cloud)](image)

**Figure 2:** SmartData Architecture (PIH on the cloud)

### 4. PROOF-OF-CONCEPT DEMONSTRATORS

#### SCENARIO 1: POPULATION ESTIMATION

This application demonstrates how encryption based arithmetic can be used to provide local authorities, security companies or other interested parties with data about people present in households in a given area. Accurate information about presence in an area can help local authorities to plan for transportation, security, utility services. Today’s sensing techniques allow to track when people arrive and leave their homes, but people in many households would feel uncomfortable sharing this information external companies, especially as there is usually no guarantee that information is not handled at an external company in unencrypted form. In this scenario, we will demonstrate that it is possible to obtain the aggregate view of the number of people present in a region without revealing the number of people present in any individual household.

The primary goal of the application is to show a possible use of privacy preserving arithmetic algorithm. Generally, it is difficult to convince potential users of the technology to its benefits, since the algorithm itself can only be demonstrated in abstract form. Showcasing a scenario where the concerns of individuals at home about sharing security critical information
(whether anyone is currently at home) are immediately obvious, but which also allows information consumers to obtain valuable information (how many people are currently within an area) provide a tangible use-case and helps to explain to potential adopters of the technology how it could be applied to their needs.

This demo integrates technologies from the following partners:

- UCAM: the PIH core platform (device)
- EURC: privacy preserving aggregation
- INTAMAC: in-home occupancy information
- FOKUS: application

**DESCRIPTION**

The underlying privacy preserving aggregation algorithm allows the data aggregator to collect individually encrypted data (number of people present in any individual household) and aggregate them (the aggregation operation is actually a simple sum) without the need for decrypting each of these values. The aggregator will only have access to the resulting aggregate value in unencrypted form. Hence, individuals will not reveal their own data but will contribute to the collective computation.

![Figure 3: Demonstrator screenshot](image)

**Occupancy data from cloud platform.** Intamac uses event data stored in their database to determine whether a property is occupied. This data is obtained from motion detection sensors that are registered within a property. Door contacts and motion detectors are used to generate the events, which are then passed to the cloud platform via a registered gateway / panel. The sensors detect movement within the property and this can be used to determine if the
property is occupied. The gateway / alarm panel status information can also be used to determine if a property is occupied. If a system is armed then it is assumed that the property is unoccupied. Based on pre-defined trigger rules, the cloud platform will determine whether a presence status change has occurred and signal the status change accordingly. The implementation allows the application to obtain data from multiple properties so that it can be aggregated.

**Implementation**

In the initial demo, we will demonstrate the underlying privacy preserving aggregation building block whereby two PIHs will encrypt data from a locking/presence detection source and encrypt them prior to sending them to a third party, namely the “aggregator”. To demonstrate integrateability with existing services, information for a small set of test households will be encrypted in the same way and be provided to the aggregator. The aggregator will perform privacy preserving arithmetic to compute the sum of these values without discovering the individual values set by each source. The demonstrator architecture is illustrated in Figure 4.

![Figure 4: Population Estimation Demonstrator](image)

The demonstrator uses the Intamac ensoCloud platform to provide data collectors for the occupancy data. The ensoCloud platform is used by organisations in various industry sectors including set-top box, Security, HVAC, Automation, etc. to connect their devices to the Internet of Things. Intamac set up presence detection rules within the ensoCloud Rules Engine for use signalling the aggregator. Secondly, Intamac provides a number of sensors along with a hub to which they communicate. The hub connects the sensors to the ensoCloud via the Internet.
Figure 5 illustrates the Intamac’s ensoCloud that consists of the following components:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ensoAgent</td>
<td>Intelligent, agnostic embedded software that runs in a hub. It acts as a communication hub between the devices and the ensoCloud platform.</td>
</tr>
<tr>
<td>ensoCloud</td>
<td>Scalable, secure, open API services platform.</td>
</tr>
<tr>
<td>ensoAPI</td>
<td>Open API Library to facilitate rapid integration with ensoCloud. It is to this component that Intamac will add Data Collector APIs for use by the Context Brokers.</td>
</tr>
<tr>
<td>ensoVideoAgent</td>
<td>Embedded software to support integration for video devices such as IP Video Cameras.</td>
</tr>
<tr>
<td>ensoInterface</td>
<td>Remote control consumer applications and user interfaces. This component will not be used in this project.</td>
</tr>
<tr>
<td>ensoEnterprise</td>
<td>Business management, administration, analytics and reporting.</td>
</tr>
</tbody>
</table>
**DEMONSTRATION SHOWCASE**

As the application operates as a demonstrator for the concept of privacy preserving arithmetic and not as a commercial service, some features have been introduced that would not be present in an operative installation.

Most notably, two demo PIHs have been set up to allow the change of the occupancy state in a demo setting, to avoid the need to wait for arbitrary changes in one of the test households.

For these ‘demo households’ the presence status is changed by using buttons on a keyfob to directly signal ‘at home’ and ‘away’ status. The rule system employed within the ensoCloud uses additional means to determine presence or absence. For example, users might forget pressing the ‘away’ button when leaving the house and there might be more than one person in a household, so pressing ‘away’ on one keyfob might not sufficiently signal an empty house. In a real household, the fact that no active keyfobs are within detection range of the hub is a more reliable indicator of non-presence. In a demo situation, however, walking away a sufficient distance from the demo setup to trigger a status change is impractical, hence the direct triggering of a status change with a keyfob button.

For the demo situation, the demo devices signal their status to aggregation when a status change occurs. In a real application, this would be a security/privacy issue, as an external observer would only need to know the occupancy status of a household at any time and then could just count the messages sent to the aggregator, without needing to decrypt their content. In a real application, this could be easily handled. As occupancy status in an area is not time critical, various solutions could be employed - they current status could be sent at fixed time intervals (just once per hour), at random time intervals or only on request by the aggregator.

In the demonstration setting, the software will be set to send status changes immediately to the aggregator to provide faster feedback on user or demonstrator interaction. A status display that updates an hour after the status changes might be realistic, but is unsatisfactory in a demo context.

**DIFFERENCES TO Y2 DEMO**

The basic component of this demonstrator (privacy preserving arithmetic) has been shown as a technical demo in Y2, demonstrating the functionality in an abstract way. Based on this, the showcase demonstrator has been developed, building an application to put the technology into a realistic environment. The technology has been combined with the Intamac enso system, allowing direct local interaction with sensors as well as communication with presence events from the ensoCloud to demonstrate an end-to-end application.
Figure 6 shows the elements of the UCN system architecture utilized by the population estimator demonstrator.
SCENARIO 2: SMART TV MOVIE RECOMMENDATION

This scenario develops an IPTV recommendation system that can work with different levels of knowledge about the user movie preferences and the user context. The user’s PIH contains several personal data sources, including a private store of their movie reviews and ratings as well as activity and location trace. The user accesses a movie preference service that provides a set of movie recommendations on two occasions: after a long day at work interacting only with colleagues, and on a relaxing holiday.

Based on the information that the user is willing to share with the service, the service is able to deliver better movie recommendations taking into account both the user’s preferences and their current context. Suggestions, recommendations and other context-aware service enablers have been developed based on user content consumption and other complementary sources of data (e.g. Facebook, IMDb). The PIH receives information from several personal data sources, stores it if needed and applies privacy and secure algorithms according to the user needs; under the control of the user different encryption and access control rules may apply to both resources and data, depending on the type of consumer (i.e. applications, service enablers, final end-users).

This demo integrates technologies from the following partners:
- UCAM: the PIH core platform (CubieTruck)
- EURC: multi-user searchable encryption (MUSE)
- PTIN: Sticky Policies Access Control and Encryption Scheme (SPACE), IPTV platform
- INRIA/TECHNICOLOR/PTIN: recommendation algorithms (generic & targeted)
- INRIA/NOTT/CAM: user activity inference

DESCRIPTION

This scenario demonstrates how we built an IPTV recommendation system adapted to users’ different privacy settings.

No access. The user denies access to the service to any data (based for example on its source). The system will then provide generic recommendations that are not personalised or contextualised. We will use the algorithms from TECH/INRIA (as well as PTIN commercial IPTV algorithm) that mines public databases of movie reviews to identify similarities among movies and present a list of similar movies. This is considered the default situation.

Cloud access to data resources. The user provides access to the history of all or some available data, but both data and access rights are encrypted before leaving the PIH (PTIN – SPACE) for remote processing. Data is sent securely to a private cloud (at PTIN premises) and to a middleware platform where authorised services, as recommendation and notification services, can subscribe and be notified of context changes in real-time and also have access to historical information if needed. Access control policies will “filter” the kind of information available for each service/consumer - e.g. the TV movie recommendation service will only have access to the TV consumption user data. With access to movie ratings and mood, the
system will be able to filter similar movies based on user preferences, using Matrix Factorization on top of the list of movies presented in the previous case (TECH/INRIA).

**Local access to data sources.** The user provides access to all available data – specifically media preferences, user activity logs, and location history – but for local processing only. That is, the consumer must provide computation to the PIH-gateway device for it to execute locally. The user is then able not only to exercise control over the initial request (based, for example, on checking simple properties such as the consumer’s identity, data sources accessed, and historical interactions with that consumer both personally and by other members of the user’s social network), but also over exfiltration of the result of the computation. In this case, the system can tune the recommendation to the user context.

**IMPLEMENTATION**
This Demo presents the algorithms from TECH/INRIA that identify movie similarities and tagging integrated with the Altice Labs (former PTIN) IPTV platform and recommendation system, with different levels of permissions. Data necessary to the algorithms is collected from different sources, namely TV consumption from both IPTV Boxes and mobile devices/gateways and user preferences (movie rating from IMDb and Facebook and mood from a PTIN application called Guider – a TV content agile-search mobile application).

Altice Labs will demonstrate SmartTV, a recommendation system based on the TV activity of the client retrieved from the set-top box (which includes real time consumption of content, as well as the visualisation of videos and automatic recordings – available for 7 days). SmartTV also takes into account information collected from IMDb, the preferences of the user (from Facebook) as well as information collected from other users. Using the combination of TECH/INRIA/PTIN movies and series classification, similarity and collaborative algorithms, the Demo illustrates the following:

- Recommendations of Live and Recorded TV programs, that the user might want to watch;
- Identification of other programs related to a specific program, that the user might want to watch;
- Search optimization that orders the results according to what the user might want to watch first;
- Highlights of new and trending programs that might interest the user.

Users can choose which TV activity they share with the system. They can choose to collect the data or not, depending on whether they activate the collection application or not. After collection, the information is stored in the PIH in house or/and in the cloud. Here, users need to authorize the access to that information or not. And even if they do, they can encrypt it and attach a SPACE policy, guaranteeing that only systems that conform to the policy will have access to the decrypted version. This Demo also demonstrates how to configure the policies in SPACE and how to give access to the PIH Store, the place where the information is stored.
The STBs provided by PT IPTV MEO service are based on Ericsson Mediaroom (formerly Microsoft Mediaroom). The application that records the viewings belongs to the application layer which runs on top of the Mediaroom software. From this onwards, it has 2 paths:

- If the user has a PIH in house, the STB finds it automatically and sends the watch logs to the SmartData platform through it. In principle, the logs leave the house encrypted, so when they arrive to the platform, they are already protected (in this way we give total freedom to the user to choose which level of trust he wants to have with the cloud provider);
- If there is no PIH in house, the STB send the logs directly to SmartData over a secure channel, where they are encrypted on arrival, according to the policies specified by the user (in this case the user chooses to trust the cloud provider but can still decide on the level of trust with third parties).

Please note that in case of UCN, some services as the TV recommendations are also provided by the cloud provider (as they have been described as part of the service enabler module in SmartData) but they behave as third party services, meaning service enablers may be explicitly authorised by users to access their data if they choose so similarly to third party applications. So, in fact the core of the SmartData (defined in Figure 2 as the context broker) is the one responsible for interacting with the privacy modules to encrypt data (if needed) together with the control access rules defined by the user.

Anyway in both cases described above, information is stored encrypted on a PostgreSQL database, for lifelong storing, and routed to services approved by the user (consumers). If the user gave permission to the SmartTV SE to collect its data, it is also routed to a Cassandra big data database, where it is maintained for a short life (e.g. TTL for EPG is one month). The mobile applications connect to the user PIH but can also connect directly to the PIH on the cloud.

![Figure 7: SmartTV high level architecture](image)

The SmartTV Recommendation (as part of the service enabling layer) can be divided in two modules that sit on top of the data provided by the SmartData core platform. In the first
module, the SmartTV API, all the data concerning a user is processed and given access through a REST API, including the most watched channels, programs and the trends. This data is used by SmartTV Recommender that also provides a REST API with the services described in the recommendations chapter.

By using additional Collaborative Recommendations, we also use data collected anonymously from all PT IPTV users. This data is also stored on the Cassandra database and made available through the SmartTV API.

In SmartData, each input from the user (STB, TV Mobile,...) has a specific policy. So, according to the user’s decision, he may give access only to a subset of his data.

DIFFERENCES TO Y2 DEMO

This demo adds the Search and Highlights Recommendations. It also adds the collaborative recommendations, made possible by the implementation of a system that mines (anonymously) real PT IPTV clients. The full integrated system is currently under evaluation with Beta testers.

The integration with the PIH (CubieTruck version) is also complete, with the storage of collected data on PIH Store and subsequent authorization mechanisms. It is also new the interface to define policies for SPACE and the integration with the new Inria/Technicholor program classification module.

PLACEMENT IN ARCHITECTURE

Figure 8 shows the elements of the UCN system architecture utilized by the Smart TV movie recommendation demonstrator.
The modules involved in this scenario are:

**Data collectors**
- IPTVBox – collects information from the TV user content viewing
- IPTV Mobile – collects information from the TV mobile user content viewing
- EPG – Electronic Programme Guide: collects the TV programs transmitted in the current and past week.
- User preferences and Social Networks – information from the Guider mobile app ("mood rating"). e.g. Boring, Facebook, IMDb

**PIH**
- PIH Store – stores user data at the user’s premises
- PIH Gateway – ensures communication with the cloud and controls data access

**Security & Privacy Service Enablers:**
- MUSE (Multi-user Searchable Encryption) – To encrypt metadata information
- SPACE (Sticky Policy Access Control Encryption) – to encrypt the collected data

*NOTE: Both MUSE and SPACE may run in the home-PIH or/and in the cloud-PIH*

**SmartData**
- Context Broker – includes data aggregation, routing and communication with the privacy modules if needed

**Recommendation Service Enablers:**
- TV Recommender/Profile Manager – to suggest and recommend TV content
- Structured Recs – Recommendations based on public ratings (i.e. IMDb)
SCENARIO 3: ASSISTED LIVING SERVICES

The main goal of this application is to assist elderly and diseased people and monitor their well-being status frequently. Users can monitor themselves, measure basic health signals (e.g. heart rate, blood pressure, weight) and log this information using an app either on their mobile devices or on their TVs. Care taker can then access this information locally or remotely to detect when patients need to visit the hospital or receive a visit from a nurse or a doctor. Care takers can also take the measures on behalf of the users in case needed (e.g. on their regular home visits). Since health condition is extremely sensitive information, users are only willing to share this information if it can be safely shared, exclusively with the people of their choice (e.g. doctors, care takers, family).

This scenario also considers a personal training use case. Focused mainly on elder people, the application allows the care takers to define exercises to be done in front of the TV. These exercises explain on a video form how to perform the movements and the application tracks which ones were made or skipped. The user has also the possibility to perform exercises suggested by the platform based on the measures collected with the sensors.

It is planned that in the future some users will be allowed to define their own set of exercises and connect with their gyms to have some professional guidance.

This demo integrates technologies from the following partners:
- UCAM: the PIH core platform (CubieTruck)
- EURC: multi-user encrypted search (MUSE)
- PTIN: sticky policies access control and encryption (SPACE); Smart AL platforms
- PTIN, INTAC: Sensors

DESCRIPTION

This scenario demonstrates how the Assisted Living system was built to allow different access rights to users’ biometrics and how it can help to control and improve the health of its users. These data is particularly sensitive for remote access, so SPACE is used to encrypt the data before leaving the PIH. Data is then sent securely to a private cloud (at PTIN premises) and to a middleware platform where services, as the AL services, can subscribe. Access control policies “filter” the kind of information available for each service/consumer. For example, let’s assume that the following attributes are required for accessing data originating from “device 1”:
- Having the role admin;
- Belonging to the group physician.

And, for accessing only the weight measurements data from “device 1”:
- Belonging to the group physician.

In this case, physicians are able to access all weight measurements from “device 1”. By taking the role “admin” they may also access all the measurements from that device but only the data provider/user will have access to the rest of the data it encrypts (coming from all the other
devices). A consumer is able to retrieve the encrypted data from the cloud and decrypt it. It has to send the policy attached to the data to a Trusted Authority (TA), which will evaluate the policy and generate a key for decryption. As the data was encrypted by the policy attached to it, any change to the policy or data will result in failure to decrypt.

**IMPLEMENTATION**

This demo is based on a pre-product from Altice Labs-PTIN called SmartAL (Assisted Living). This solution has been enhanced with is a set of notifications and advice services based on well-being sensor information (blood pressure, blood sugar, weight, etc.) collected regularly from users (via the set-top box or a mobile GTW – phone, tablet). This information is sent securely to a private cloud in PTIN premises using SPACE (located in a simplified version of the PIH, i.e. Raspberry Pi).

The demo illustrates what and how can a care taker manage his patients (using the backend application). This includes checking their evolution and defining intervals for their measures (assisted by the predictions produced by the system).

Other important part is the training use case. In the demo it is possible to see how a care taker manages patients’ training sessions and how the patient uses the TV to execute them and checks the recommendations given by SmartSense.

This system is divided in a similar manner to the SmartTV recommendation system. The main application also runs on the IPTV STB where the user can access all the functionalities, although there is also a mobile version running on smartphones and tablets. The smart devices/sensors that the user has in house, that allow him to perform the measures (heart rate, blood pressure, blood sugar level, weight, etc.), communicate directly to the PIH through Bluetooth (in the specific scenario of Altice Labs/PTIN a Raspberry PI is used as an extension of the PIH to ensure the Bluetooth communication). There, the measures are encrypted with the help of the access control algorithm developed by PTIN (i.e. SPACE) and, if authorized by the user, they are published to the SmartData platform in a secure manner.

In addition, there is a companion mobile app which communicates with the PIH and allows manual insertion of measures (in case of miscommunication of the devices) and control of “to-do” tasks by care takers to keep them always updated with the tasks they have to perform on behalf of the user.

In the SmartData there is a dedicated Service Enabler - SmartSense - where all the user data is stored. This SE is also responsible for analysing the evolution of the user, generate warnings of values out of normal, predict intervals to help in diagnosis and create exercise suggestions (special feature for the personal training service).

The SmartAL application sits on top of the SE and manages users and their relationships. The backoffice offers tools for users, doctors and care takers to control the overall system. The autonomous user can check his evolution (complementing TV and mobile) and define trainings. Care takers can check their patients’ evolution, control intervals for each measure (assisted by predicted values) and define training plans.
**DIFFERENCES TO Y2 DEMO**

This demo increments the functionalities of the backend and adds the prediction of values to assist carers.

A new personal trainer use case has been introduced. New recommendations of training videos have been developed and a new training application for TV has been implemented.

**PLACEMENT IN ARCHITECTURE**

Figure 10 shows the elements of the UCN system architecture utilized by the Assisted Living demonstrator.
Figure 10: Assisted Living demonstrator coverage of architecture elements

The modules involved in this scenario are:

**Data collectors**

Sensors – Well-being devices that analyses users’ vital signs and health data (blood pressure device, scale, etc.)

**PIH**

PIH Store – stores user data at the user’s premises
PIH Gateway – ensures communication with the cloud and controls data access
Security & Privacy Service Enabler:
  SPACE (Sticky Policy Access Control Encryption) – to encrypt the collected data

**SmartData**

Context Broker – includes data aggregation, routing and communication with the privacy modules if needed
Recommendation Service Enabler
  AL Recommend and Notify – module to notify the users about threshold violations and dangerous variations on his measures, as well as recommend/suggest exercises based on learning and predicting
SCENARIO 4: HOME NETWORK MONITORING AND DIAGNOSIS

The goal of this application is to assist users and their ISPs in diagnosing network performance problems in the home. Most users are not experts in networking and consequently are helpless when applications perform poorly. Their only resort is to call the ISP, which has very little information of what is happening within the home network. UCN will provide tools for users and ISPs to visualize the performance of online applications, detect when performance is poor, and diagnose whether the problem lies in the home wireless or the access link. The implementation of this scenario requires fairly sensitive user data (e.g., applications running on the users’ device, packet headers). UCN will enable this application while preserving users privacy, because all the data is safely stored on the PIH.

This demo integrates technologies from the following partners:
- UCAM: the PIH core platform (device) and Irmin for data storage
- INRIA/TECH: home gateway data collection, prediction of WiFi quality on Web QoE, visualization
- INRIA: diagnosis of home wireless versus access problems
- TECHNICOLOR: home gateway data collection, data analysis and visualization

DESCRIPTION

The PIH will collect information from the home gateway to assist with network diagnosis. The PIH will host an application that processes data collected from the home gateway to: (i) identify whether performance problems lie in the home wireless network or the access link; and (ii) predict the impact of the quality of the WiFi on Web QoE. Users can access a web page with a dashboard that is running on the PIH. The dashboard will show the inferred quality per device of a given user. We will show whether the bottleneck is in home or on the access.

IMPLEMENTATION

We demo the home gateway sending data to the PIH (in a CubieTruck). The home gateway is running the collection and diagnosis methods described in D2.3 (Sec. 3.3.3) and the prediction of Web QoE from Wi-Fi quality metrics described in D3.4. The system consists of two data collectors running on the gateway, the PIH data storage component, and a web interface that displays the results to the user, as displayed in Figure 11.
Wireless data collection. We initially developed the wireless data connector using the scripting language Lua to invoke the iw command line program for every sample. We found that the overhead of creating a new process multiple times per second was imposing a limit on the sampling rate, as well as some system load as shown in the previous section.

For the full system implementation, we decided to port the wireless data collector to C and use the Netlink library instead of the iw program. The Netlink library is used to communicate with the Linux kernel through the Netlink protocol, an inter-process communication mechanism. Notably, the wireless subsystem of the Linux kernel exposes many relevant metrics through Netlink.

The wireless data collector outputs two sets of results to configurable locations on the file system. The first set consists of all the samples in their raw form, and is used primarily for troubleshooting and offline analysis. The second set includes aggregates such as the average PHY rate and the change in packet counters over an interval of 10 seconds. It also includes the throughput and estimated link capacity over the same interval.

Access link data collector. We use the liboping library to ping the default gateway. The collector outputs, to a configurable location on the file system, aggregate results every 10 seconds, consisting of the maximum, minimum and median RTT for the interval, the IP address of the destination of the pings, and a timestamp.

Data upload to PIH. We upload the results from the two collectors to the PIH using a simple shell script running on the gateway. The script polls the directories where the collectors output the results and upload each file to the PIH via HTTP before deleting it.

The PIH offers an HTTP API for both uploading and downloading the data. The application can upload a JSON file to any URL, including any number of components (e.g. /a/b/c.json). The new file will be created even if some of the components do not exist yet. If the file
already exists, it will be simply overwritten. After uploading a file, it can be retrieved at the same URL. There are also two endpoints for each path components: /a/list returns a list of files in directory a, while /a/all returns all the files concatenated as a single JSON object.

We designed the URL structure to provide a limited support for time range queries, which are very useful for displaying time series. We name each file with its time stamp (in seconds since the Unix epoch), and we upload it to a directory obtained by dividing the time stamp by 600 and keeping the integer part. Each directory contains 600 seconds (10 minutes) of results, which is an adequate granularity for our display queries.

**Web-based User Interface.** The UI consists of a web page that is served from the PIH itself and uses AJAX to load and refresh the data dynamically. The web page implements the detector and the predictor logic, rather than the gateway: we did this choice because it is easier to update than the code running on the gateway, and we want to be able to tweak the detectors/predictors if needed. The page presents to the user a plot for the wireless link, one for the access link, and Web QoE predictor.

**DIFFERENCES TO Y2 DEMO**

In Year-2 we presented a demo of the algorithm to detect home wireless versus access link bottlenecks and sending data to the PIH. As discussed in D2.3, the original algorithm presented performance issues when running in a larger deployment, so this year we’ll present a new algorithm that is based on our models to predict wireless capacity combined lightweight pings of the access link. This demo adds the wireless capacity estimation (D2.3) and the prediction of the wireless quality on Web QoE (D3.4). The home gateway data collector is now more efficient because it is implemented in C and it is integrated with the PIH store. The Web interface has been updated to show the results of the new algorithms/models.

**PLACEMENT IN ARCHITECTURE**

Figure 11 shows the elements of the UCN system architecture utilized by the Home Network Monitoring and Diagnosis demonstrator.
Figure 11: Home Network Monitoring and Diagnosis demonstrator coverage of architecture elements
5. **SUMMARY**

The objective goal of the UCN project is to “provide a new paradigm leveraging user information at large to deliver novel content recommendation systems and content delivery frameworks. UCN recommendation and content delivery systems will leverage in-depth knowledge about users to help them find relevant content, identify nearby network resources and plan how to deliver the actual content to the appropriate device at the desired time. These systems will additionally account for influences from users’ social networks on their content consumption. The goal of this project is to design a UCN system architecture for user-centric connected media services.”

For this, UCN provides an architecture that supports:

- **Understanding user context:**
  This is the task performed by the data collection tools, which retrieve data from the user devices, store this information and derive contextual information from it.

- **Profiling and predicting user interests:**
  Based on the information collected, predictors and recommendation engines attempt to anticipate user interests.

- **Service personalization:**
  As a result of the assumptions and predictions resulting from the preceding point, content needs to be offered to the user in a manner suitable for this user in the current context.

The four demonstrators developed demonstrate the suitability of the architecture and its implementation to achieve the goals set forth for UCN.