# D3: Portal and Access Protocol

| | |
|---|---|
| **Project Number:** | **507442** |
| **Project Acronym:** | **MHP-KDB** |
| **Project Title:** | **The MHP Knowledge Project** |

| | |
|---|---|
| **Deliverable No.:** | **D3** |
| **Title:** | **Portal and Access Protocol** |
| **Version No.:** | 1.0 |
| **Related to Work package:** | WP 3 |
| **Type of Deliverable:** | Report |
| **Dissemination level:** | PU |
| **Authors:** | Simon Pfeiffer (UDE), Johan Wittebolle (TCL), Christoph Brendes (IRT) |
| **Due date:** | 31.05.2004 |
| **Delivery date:** | 16.07.2004 |

**Abstract:**
This document describes the access protocol for the web portal and database of MHP-KDB. It investigates what kinds of users exist, what privileges they have and what obligations there are for using the database. The document also sketches the basic user operations of the knowledge database as well as the structure of the database itself.
The main objective of this text is to provide a conceptual description of the portal, the database and its access protocol rather than giving a technical overview.

**Keyword list:** MHP, DVB, KDB, Database, KDB, Portal, user categories, roles and rights, entity relationship diagram

**Table of Contents**

1. Introduction 3
   1.1 Scope of this document 3

2. Overview over the MHP-KDB portal 4
   2.1 Static Part 4
   2.2 Dynamic Part 4
   2.3 Automated Testing Environments 5

3. Database structure and functionality 6
   3.1 Database Structure 6
   3.2 Functionality of the database 8
      3.2.1 Login 8
      3.2.2 Adding an Entry to the Archive 8
      3.2.3 Retrieving an Entry from the Archive 9
      3.2.4 Ontology 10

4. Access Protocol 11
   4.1 User categories 11
   4.2 Rights & Roles 12
   4.3 Restricted Actions 14

Annex1: Entity Relationship Diagram of the Database 15

# 1. Introduction

Work package 3 delivers a number of tools and supporting facilities in order to gather and provide design knowledge for creating, assessing, validating and testing MHP based applications. Part of work package 3 is the back end infrastructure and the portal that grants access to the KDB content and which displays the results of the project.

## 1.1 Scope of this document

This document describes the access protocol for the MHP-KDB database, i.e. what kinds of users exist, which privileges they have and which obligations there are for using the database. It also sketches the basic user operations of the knowledge base.

The document aims at providing a more conceptual description of the portal, the database and its access protocols rather than giving a technical overview.

The following items are investigated:

- Brief overview over the portal
- User categories
- Rights and roles
- Usage of the database
- Overview of over the KDB structure

# 2. Overview over the MHP-KDB portal

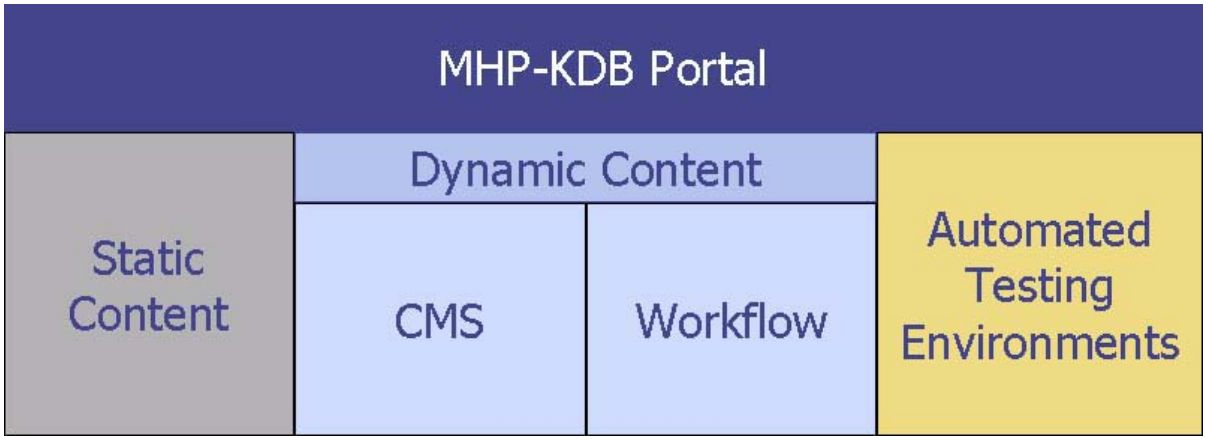


**Figure 2-1: Conceptual structure of the portal**

The structure of the portal, which is displayed in **Figure 2-1** was described in detail in deliverable D2 Part A.

All the content of the portal will be accessible via internet with a standard web browser. The portal will provide a single entry point for all of its components as described below. There will be one central login which will be used to determine the privileges of the user for all areas of the portal.

The first “prototype” version of the portal already offers static content and includes working areas for the various partners which allow for uploading and publishing files. This prototype is currently available under http://www.mhp-knowledgebase.org.

## 2.1 Static Part

There is a static HTML component that presents various MHP-related topics and information to the public (e.g. MHP news, guidelines, FAQs, etc.). This element of the portal will contain more general information that is categorised with respect to the expected user groups (e.g. there will be branches that focus on topics that are interesting for TV consumers, branches for manufacturers and so on). This portion of the portal also comprises all the public deliverables, e.g. those of work package 4, especially the MHP guide. Additionally, the project will be introduced and the user is informed about the content that he will be able to see if he registers with MHP-KDB, thus this part also works as a "teaser" for the dynamic part of the KDB portal and the automated testing environments which will be accessible via the portal.

## 2.2 Dynamic Part

The dynamic section, which is based on the actual database, will consist of two parts: Firstly, there will be the web access to the knowledge database with the possibility to perform queries on the database and enter new content to the database. This will be the main section of the portal where most of the content, which is produced in the course of the project, will be presented. It includes that part of the WP4 output which is not presented in the static component (all the info about tools, source code, applications, etc.) and all the data entries created by the knowledge-base users etc.

Technically, this section will be supported by a content management system that is also responsible for managing the front-end and a document management system taking care of search functionality and persistent storage as a back-end.

Secondly, there will be a workflow component facilitating the cooperation of the project partners. Its degree of sophistication will be adjusted to the requirements of the project, i.e. with respect to the level of collaboration between the partners when generating content.

### 2.3 Automated Testing Environments

The portal will also offer access to automated testing environments that enable application programmers to test their applications without having to set up their own (costly) environment. It is planned to offer two different kinds of automated testing environments (ATE): Normally, the ATE is accessible via the KDB web portal which allows the programmers to configure a test suite that meets their requirements and have their applications tested on the server side. The ATE will reside on the corresponding service provider's site, i.e. they will only be linked to via the portal. Alternatively, test tools may be made available for downloading which allow carrying out some basic tests of applications at the client side.

As mentioned above, the login to the web portal will also serve as an account for the testing environments. In order to avoid constant synchronisation of user privileges between ATE sites and the KDB administration, authentication has to be handled at the ATE provider's site using the [username, password] combination as transferred from the portal. At the KDB site it will only be checked whether a user has the right to use an ATE at all. For now, every registered user has the permission to use the link to an ATE. In the future, there can be an additional privilege called "ATE", if required.

If the user is rejected at the ATE provider's site, appropriate error handling needs to happen there. The process of accessing an ATE via the KDB portal is depicted in **Figure 2-2**.

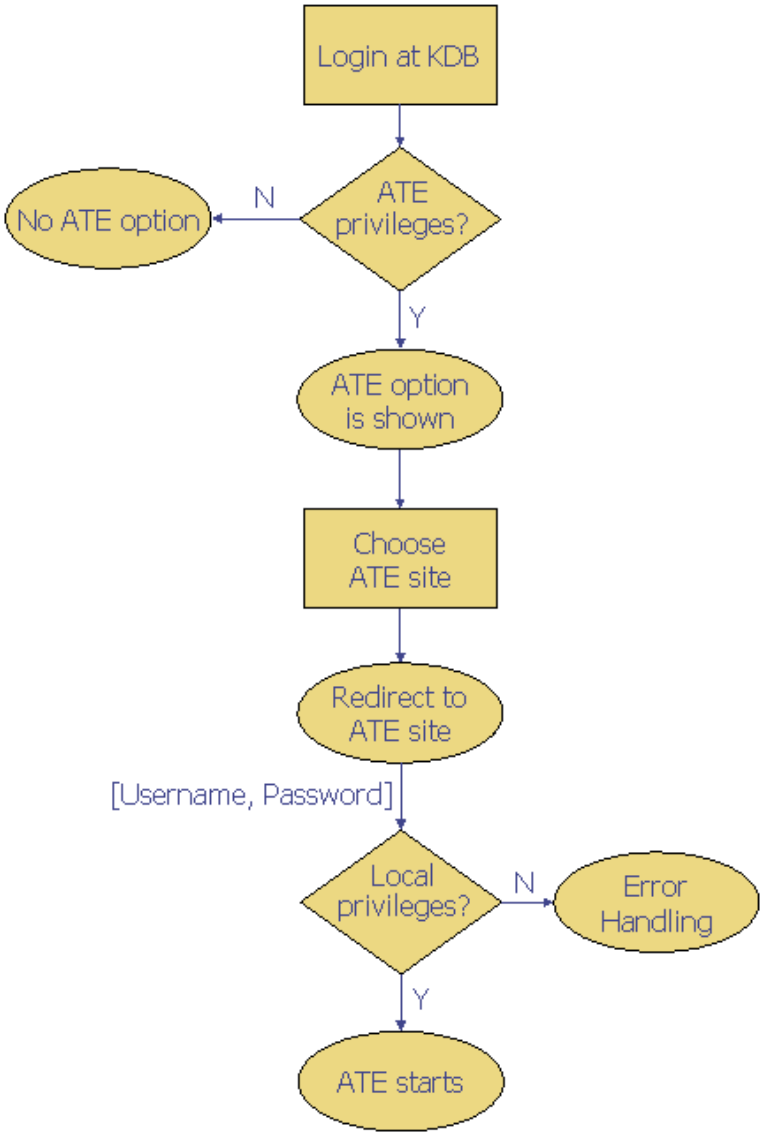


**Figure 2-2: Logging onto an ATE**

# 3. Database structure and functionality

This chapter further examines the dynamic part of the MHP-KDB portal. Here, the data types of the database are described and basic functionalities like login, search mechanisms, etc. are depicted.

## 3.1 Database Structure

The entities contained in the database have already been mentioned in deliverable D 2 Part A. This document (D 3) contains a refinement of the entities including their attributes and associations. Since new requirements will certainly arise during the course of the project, this structure is subject to change. **Annex 1** contains the corresponding entity relationship diagram of the system. For the rest of this paper, the term “document” refers to one of the data types displayed in **Table 3-1**. There, the data types (Document Type) are described briefly, the related attributes are listed and the links to other data types (Associations) are given. A ‘+’ with the associated data type indicates that at least one link is required (1 ore more) whereas a ‘*’ means that the association is not mandatory (0 or more).

The following peculiarities should be noted: The "issue" type is the central data type of the system. It is the only data type without creation restrictions (see section 4.3 for more details) that exists on its own. Any other data without creation restrictions needs to be associated with an issue or another document that is transitively linked to an issue (i.e. solutions, comments, source code and attachments).

| Document Type | Description | Attributes | Associations |
| --- | --- | --- | --- |
| | | | |
| **Issue** | Documents of this type describe an MHP related issue of any kind | Long Description: Text<br>Issue-Type: (probable solution type of the issue e.g. Standard modification, test suite modification, Decoder related, application related, guideline)<br>Aggregated Priority (computed from the Assessment-Association)<br>Category+: Category-type<br>Status: (draft, open, settled, internal open, etc.)<br>Error Type (performance, QoS, Exception)<br>Keywords: Text | Assessment+: Person-Link (Priority)<br>Application*: Link<br>Decoder*: Link<br>Tool*: Link<br>Attachment*: Document-Link<br>MHP-Corrigenda<br>Standard*: Link<br>Category+: Link |
| **Application** | Representation of a complete MHP Application capable of running on top of a STB | MHP-Application: MHP application standard attributes, (e.g. main class, …)<br>Category: Application-Category, e.g. game, news, interactive, production, test<br>Required-Resource*: Category<br>Signature: yes \| no | [Source-Code]: Java-Code-Link, Environment-Description<br>[Service]: MHP-Service-Link (MHP-Application-ID)<br>[ClassFiles]: Java Class Files<br>[TS-File]: Transportstream containing application (AIT info)<br>Producer: Company-Link |

| Document Type | Description | Attributes | Associations |
| --- | --- | --- | --- |
| **Decoder** | Representation of an MHP compliant STB, or other compliant device (TV-card, TV-Set,…) | Model-ID: (Name, Version,…)<br>Software-Version: Version-Number<br>Hardware-Version: Version-Number<br>Decoder-Type: (TV-Set, STB, PC-Card, Mobile-Phone,)<br>DVB-Standard: DVB-[T\|C\|S\|H]<br>VM-Version: Version-Number<br>[Font-Renderer]: SW-Component<br>Market-Status: tbd<br>Relevant-Market: Area+<br>MHP-Standard-Version: Version-Number (e.g. x.y.z)<br>MHP-Profile: (interactive-broadcasting\|enhanced-broadcasting\|internet-access) (level-1\|level-2)<br>[MHP-Stack]: SW-Component<br>Certification-Status : (yes, no, in-progress)<br>[Resource-Model]: (Memory, HD, Return-Channel, Graphics, CA/CI)<br>[Tech-Spec]: URL of the Company-Website | VM-Vendor: Company-Link<br>Manufacturer: Company-Link |
| **Tool** | Representation of MHP-Tools (Authoring, Playout, Testing, Analysing; …) | Tool-Name:<br>Version: Version-Number<br>Tool-Category: (Authoring, Playout, Testing, Analysing) | Producer: Company-Link |
| **Standard** | Document MHP Specification (ETSI-Document) | Version: Version-Number | |
| **MHP Corrigenda** | DVB internal list of corrigenda items | Version: Version-Number<br>MHP-Standard: Version-Number | |
| **Source-Code** | Java Code Fragment | | Issue<br>Solution<br>Application |
| **Solution** | Solution of an Issue | Long-Description: text<br>Ranking: tbd | Source-Code<br>Issue<br>Attachment |
| **Attachment** | Document of any Type related to other Document | Attachment-Type | Solution<br>Issue |
| **Comment** | Text Document | Status: tbd | |
| **Service** | Representation of a DVB Service containing MHP applications | Service-Name<br>Description<br>Bouquet<br>ServiceID<br>TransportstreamID<br>Network-ID | Broadcaster: Company-Link |
| **Company** | Representation of a Company | Company-Name<br>Address<br>Company-Type (Broadcaster, Producer, Developer)<br>ResponsibleContactPerson: Person<br>Membership (Part of MHP-KDB or not) | |
| **Category** | Category of an Issue | Name<br>Level | Standard |

**Table 3-1: Data Types of the KDB**

## 3.2 Functionality of the database

### *3.2.1 Login*

As one might expect, the login procedure sketched in **Figure 3-1** is very straightforward. The users are required to enter a user name and password, and after the privileges are checked they will see the first screen of the application. If no user of the given name and password combination exists, the user will be redirected to the login screen.

Depending on the user's privileges, certain menu points will be available. For example, if a user who is with a decoder manufacturer logs in, the list of possible items to add will also contain decoders (granted the project uses the more restrictive approach of adding decoders, services, etc.).

There will also be a guest login which only allows viewing the static HTML content. This can also happen implicitly by having users not logged in at all but rather having a link from the portal that leads to the static HTML pages but not the dynamic, database-driven part.

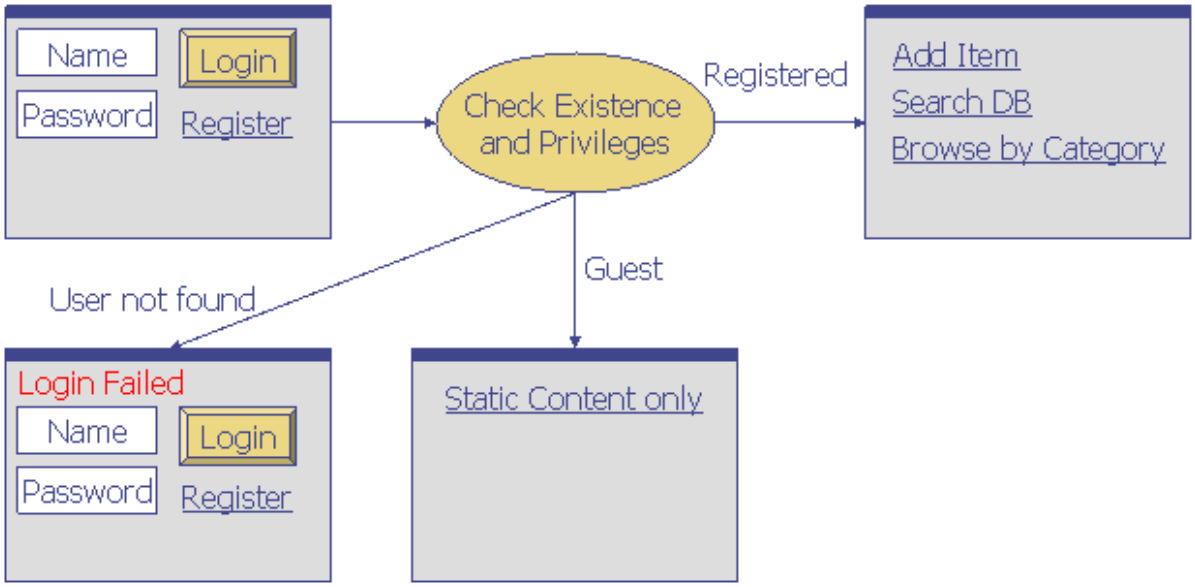


**Figure 3-1: Login procedure**

### *3.2.2 Adding an Entry to the Archive*

In order to archive a document, users need to have the corresponding privileges. After login, the user can choose between *adding a document* and *looking for documents* (either by a meta-tag search or by categories). Reviewers will have a link to their list of documents to be reviewed as well. **Figure 3-2** shows how adding an issue would work from the user perspective: After the user has chosen to add a document, the list of document types that this user is permitted to add is displayed. After choosing a type (an issue in our example), the corresponding meta-information for this type can be entered. It is also possible to upload files that will be associated with the document as attachments. Then, the user will see an updated list of issues. This can be for example a list of the issues of which the user is the author, or a list of issues associated with the same keywords the user has just entered etc. – this is configurable. However, it should contain the newly added issue in order to show the user that the addition worked.

In order to add any of the other document types that require an association to an issue, such issue has to be searched and selected first (see the next section).

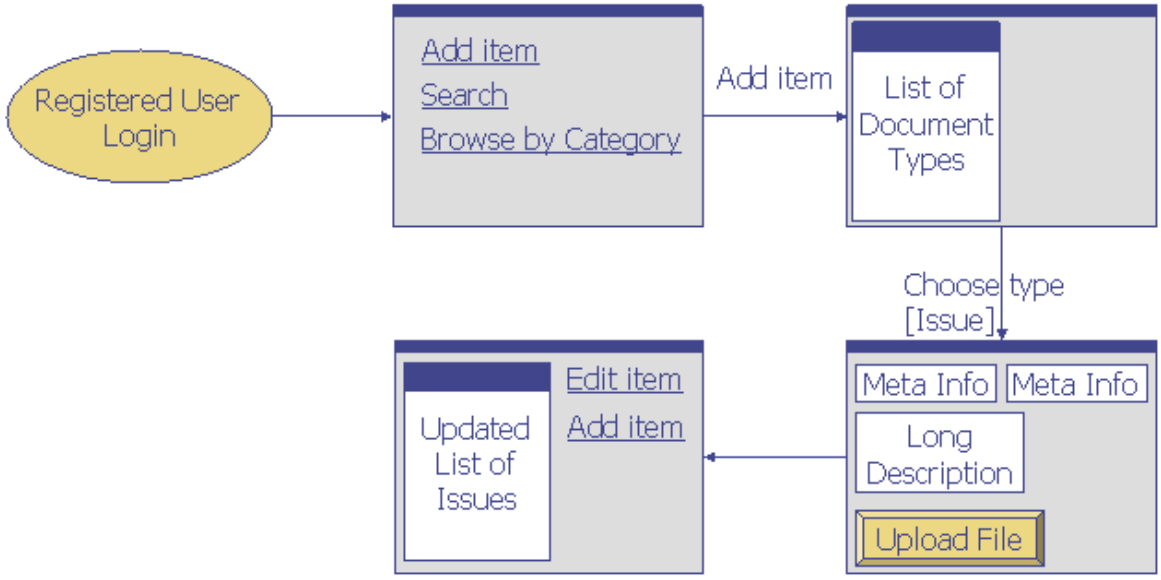


**Figure 3-2: Adding an Issue**

### *3.2.3 Retrieving an Entry from the Archive*

In order to retrieve a document from the archive, the user chooses the search option after login. The user then needs to choose the document to search for (in our example displayed in **Figure 3-3**, it is again an issue). After that, the user can add the meta-information and is presented a list of results. This list will contain the titles of all the issues matching the search request. The user can then choose which of the issues to display in detail. The detailed view will also contain the possibility to add solutions and comments to an issue.

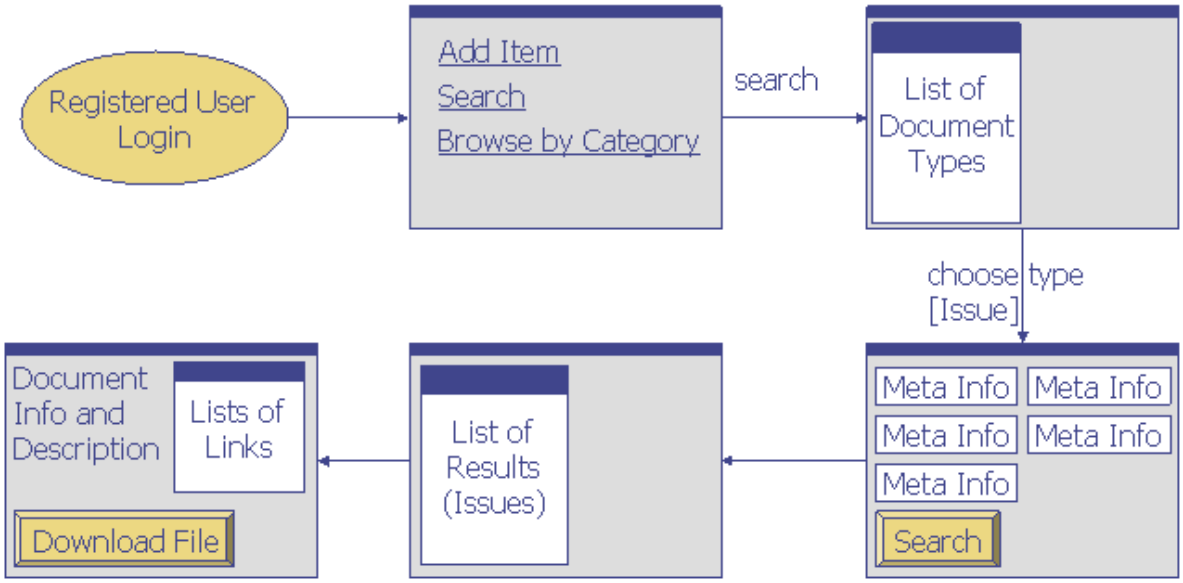


**Figure 3-3: Retrieving a document via Meta Search**

It is also possible to find documents using the *browse by category* function. Principally, this works like most commercial portals: The user navigates through hierarchically organised categories until the leaf nodes of the categorisation tree are reached and a list of documents of the chosen category is displayed. This process is depicted in **Figure 3-4**. It is of course also possible to let the user choose the document type to search for in advance and then only display a list of issues, for example.

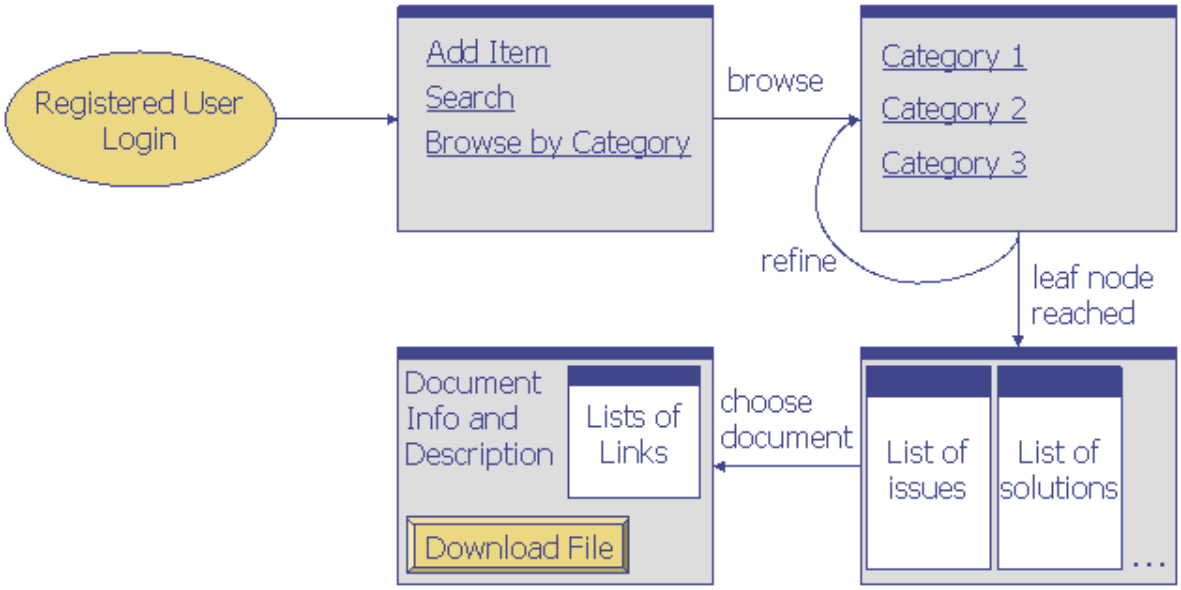


**Figure 3-4: Retrieving a Document via Categories**

### *3.2.4 Ontology*

In order to support the *browse by category* search, the KDB content (e.g. issues, solutions, code samples, etc.) has to be categorised by suitable keywords. A system of category types is being developed in WP 3 by deriving keywords from the structure of the MHP-standard itself and by considering other criteria which are not reflected by the MHP standard. The category types will be structured in a two-level hierarchy of keywords. The first level of category types has been defined as follows:

- Basic Architecture
- Transport Protocols
- DVB-HTML
- Application Lifecycle
- Application Signalling
- DVB-J
- Security
- HAVI - CSS 2 (MHP1.1)
- Graphics Video and Audio reference model
- Text presentation
- Other aspects (usability, performance, etc.)

The second level will define about ten additional subcategories, each carrying direct links to chapters of the MHP-standard. The detailed description of the category types will be part of deliverable D6.

# 4. Access Protocol

In order to access the non-static elements of the portal, i.e. any content of the actual database, the workflow component or the testing facilities, registration is mandatory. The registration procedure is depicted in **Figure 4-1**.

A user submits personal information (e.g. name, email address and company, if applicable). In order to verify whether a user claiming to work for a company actually belongs to that company, it is believed to be sufficient to perform a peer network check since the MHP community is not that big. This can, for example, be done by asking the appropriate contact persons of the companies for confirmation.

If a company does not exist, it will be added to the system after its existence has been verified via a peer check (e.g. via email or telephone). If the users do not enter a company name, they are added to the system without an associated company (and therefore without any company-specific privileges).

After the user has been added to the system, an e-mail containing the password is sent to the user who can then log in. The user is required to change his password when he logs in the first time.

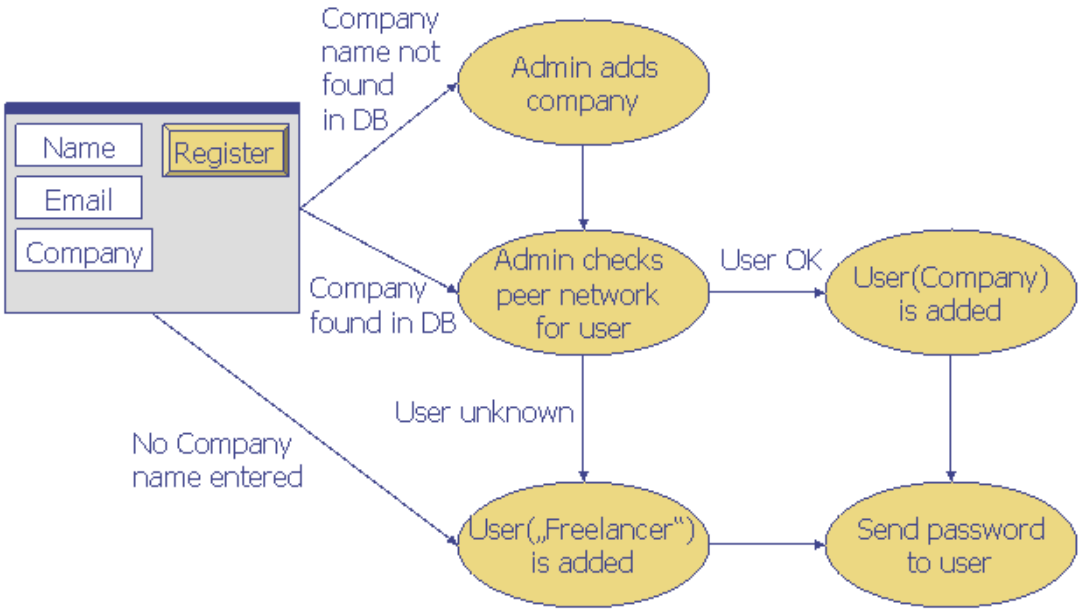


**Figure 4-1: Registration procedure**

## 4.1 User categories

The following user categories have been identified by WP 2.2 and were described in detailed form in deliverable D 2 Part A:

- Consumer
- Application developer
- MHP Student
- Broadcaster
- Middleware and / or Equipment Manufacturer
- Tester
- MHP-KDB contributor
- MHP-KDB Portal administrator

There are rules on how the user categories are linked with the roles that database users can have with regard to special restrictions or privileges. They also have an impact on the data types required for the database.

### 4.2 Rights & Roles

Up to five roles a person can have are defined. Some general characteristics of these roles are listed below:

- A role is a collection of rights.
- They are hierarchically organised.
- The rights included in a role are the same for all document types.
- Each person has exactly one role for each document.

The corresponding rights for each role are summarised in **Table 4-1** and **Table 4-2**. The alternatives are explained below.

| Role | Rights |
| --- | --- |
| **Viewer** | Read all published documents<br>Add comments to published documents |
| **Author** | Viewer +<br>Read all his unpublished documents<br>Modify all his documents<br>Publish his own documents |
| **Reviewer** | Author +<br>Read all unpublished documents<br>Modify all documents<br>Publish other users' documents |
| **Admin** | Reviewer +<br>Delete documents |

**Table 4-1: Rights & Roles, Alternative A**

| Role | Rights |
| --- | --- |
| **Viewer** | Read all published documents<br>Add comments to published documents |
| **Author** | Viewer +<br>Read all his unpublished documents<br>Modify all his documents |
| **Publisher** | Author +<br>Publish his own documents |
| **Reviewer** | Publisher +<br>Read all unpublished documents<br>Modify all documents (published or not)<br>Publish other users' documents |
| **Admin** | Reviewer +<br>Delete documents |

**Table 4-2: Rights & Roles, Alternative B**

In order to ensure that the content of the database is as up-to-date as possible, it is not intended to employ a formal review process before publishing a document, which is reflected in the rights and roles concept of alternative A. Authors are allowed to publish their content by themselves, i.e. any new content will be visible immediately. However, this does not mean that there will be no reviewing at all: The published content will be flagged as “not reviewed” in order to emphasise that the information may not have the desired quality.

After the document has been reviewed, the flag is removed. This review process entails the nomination of a formal review board which is subject to negotiation between the project partners.

If during the course of the project it becomes obvious that the quality of the content of the database is adversely affected by this practice, the publishing process will follow the review, which is reflected in alternative B. Here, authors are not allowed to publish but have to wait for the reviewer to permit publishing. Separating the reviewer and publisher role is technically not necessary, yet there may be the need to give some trusted authors the right to publish without a formal review process, hence the role of the publisher.

The review process can then look as follows:

- Each document will be reviewed by two reviewers (four eyes principle).
- There will be a workspace for each reviewer that is accessible via the portal where items are listed that need to be reviewed.
- In regular intervals (e.g. once a week), the reviewers will receive emails containing the list of items to be reviewed.
- Since the assignment of the documents has to happen according to the expertise of the reviewer, the assignment procedure will use keywords that have to be entered when a new document is submitted into the database.
- After both reviewers have marked the document as reviewed, the “not reviewed” flag is removed.
- If there are different opinions between the reviewers, it can be assumed that the conflict can be settled via communication.
- If a reviewer receives a document to review which he lacks the expertise for, he is responsible for altering the keywords and putting it back on the list of items to be reviewed.

Once a document is reviewed, deletion will only be possible by the administrator in order to ensure database consistency. In order to delete a document, this needs to be requested. Deletion candidates will be accumulated in the administrator’s workspace. In regular intervals, the administrator checks for any technical implications (i.e. related documents) and after that he checks with any possibly affected partners whether deletion is OK.

Technically, the document will not be erased at all but is made invisible by setting the corresponding flag, so it can be reconstructed at any time.

It is not intended for any user to be able to take on any role. **Table 4-3** displays a coarse-grain user type categorisation that summarises which kind of person can fulfil which role.

| Type | Description | Allowed Roles and Restrictions |
| --- | --- | --- |
| **Visitor** | An unregistered user | May only view static content |
| **Friend** | A registered user not belonging to any of the project participants’ organizations | Viewer, Author, Publisher |
| **Member** | A registered user belonging to one of the project participants’ organizations | Viewer, Author, Publisher, Reviewer and Admin |

**Table 4-3: User types**

### 4.3 Restricted Actions

Even after registration, certain actions on the database will not be allowed to everybody. There are creation and query activities which need to be restricted. E.g., creating a new entry for a decoder is allowed only for persons who are with the company which produces the decoder. **Table 4-4** and **Table 4-5** summarise the restricted actions identified so far. Furthermore, any content specific to WP 4 (guidelines, FAQs, etc.) that will be present in its special section of the portal can only be added by the project partners responsible for the corresponding work item.

| Query | Allowed for persons who are with |
| --- | --- |
| List all issues related to a decoder | the company who produces the decoder |
| List all issues related to a service | the company who offers the service |
| List all issues related to an application | the company who produces the application |
| List all issues related to a tool | the company who produces the tool |

**Table 4-4: Restricted queries**

| Creations | Allowed for persons who are with |
| --- | --- |
| Create a decoder entry | the company who produces the decoder |
| Create a service entry | the company who offers the service |
| Create an application entry | the company who produces the application |
| Create a tool entry | the company who produces the tool |

**Table 4-5: Restricted creations**

Managing the creation of these data types in this way can possibly create problems when adding new entries to the knowledge base. For example, if a user wants to add an issue related to a specific decoder, and this decoder has not been entered into the knowledge base, the appropriate link cannot be established. Thus, this approach requires a lot of discipline on behalf of the vendors of tools, decoders etc.

An alternative to the approach mentioned above could be to offer a user who adds a decoder-related issue to the database the option to choose from a list of existing decoders, and if the decoder cannot be found there, the user can enter a decoder manually. Just like non-reviewed issues, this decoder entry is then flagged as not reviewed and is required to be verified by the review board. This can be decided during the course of the project if the project partners consider the more restrictive approach as not useful. Of course, it also possible to modify the set of restricted actions at any time during the project if necessary. This would be the case, if further restrictions need to be considered or if the restrictions are found to be too rigid.

# Annex1: Entity Relationship Diagram of the Database