International Journal of Advanced Research in

Electronics, Communication & Instrumentation Engineering and Development

Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210

MULTI-RESOLUTION PRUNING BASED CO-LOCATION IDENTIFICATION IN SPATIAL DATA

Shreya Satishkumar

Student, Dept of Electronics & Communication, SRM University, India

ABSTRACT− *Security is a key concern in a wide spread network. Preserving private information is to be given due importance by all communication devices and search engines, since there is a threat of unauthorized users accessing secure information by trapping the network devices. Existing wide spread network of computers, mobile and other electronic devices does not define proper protocols neither based on user's location nor based on the end user's requirements in connecting to the network. Our proposed solution provides the most better and promising solution for a good network of plug and play Networks along with high level of authentication and authorization solutions. The proposal uses Flexi-Negotiable Security solutions that takes into account the cost and crude for such implementations along with best interoperability among the connected devices. Set of authorization policies are generated by a network manager using XACML based on the based on the available resources and the number of connected devices thus proving a reliable and secure network of devices. In this project, we are trying to incorporate a control point which will take care of controlling the devices access points. Each individual user needs to get authentication and authorization to access the resources in the network. Control point will take care of validating the request by the users. Once the users holds the authentication/authorization to access the resource in the network. They are permitted or else, no option to access the resources and they will be restricted. The authentication will be verified by the control points through a secure SOAP based web services. Our proposed system involves the above said techniques and it's associated with attribute based authentication. So that, higher designated people will be provided with more access options.*

1. INTRODUCTION

UPnP technology defines an architecture for pervasive peer-to-peer network connectivity of intelligent appliances, wireless devices, and PCs of all form factors. It is designed to bring easy-to-use, flexible, standards-based connectivity to ad-hoc or unmanaged networks whether in the home, in a small business, public spaces, or attached to the Internet. UPnP technology provides a

distributed, open networking architecture that leverages TCP/IP and Web technologies to enable seamless proximity networking in addition to control and data transfer among networked devices. The UPnP Device Architecture (UDA) is more than just a simple extension of the plug and play peripheral model. It is designed to support zero-configuration, "invisible" networking, and automatic discovery for a breadth of device categories from a wide range of vendors. This means a device can dynamically join a network, obtain an IP address, convey its capabilities, and learn about the presence and capabilities of other devices. Finally, a device can leave a network smoothly and automatically without leaving any unwanted state behind. The technologies leveraged in the UPnP architecture include Internet protocols such as IP, TCP, UDP, HTTP, and XML. Like the Internet, contracts are based on wire protocols that are declarative, expressed in XML, and communicated via HTTP. Using Internet protocols is a strong choice for UDA because of its proven ability to span different physical media, to enable real world multiple-vendor interoperation, and to achieve synergy with the Internet and many home and office intranets. The UPnP architecture has been explicitly designed to accommodate these environments. Further, via bridging, UDA accommodates media running non-IP protocols when cost, technology, or legacy prevents the media or devices attached to it from running IP. What is "universal" about UPnP technology? No device drivers; common protocols are used instead. UPnP networking is media independent. UPnP devices can be implemented using any programming language, and on any operating system. The UPnP architecture does not specify or constrain the design of an API for applications; OS vendors may create APIs that suit their customers' needs

2. SYSTEM STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major

requirements for the system is essential. Three key considerations involved in the feasibility analysis are Economical feasibility, Technical feasibility and Social feasibility

2.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

2.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3. SYSTEM ANALYSIS

System Analysis is a combined process dissecting the system responsibilities that are based on the problem domain characteristics and user requirements.

3.1 EXISTING SYSTEM:

Stipulating the requested data to the authorized user via a media has flaws and slits. In this System, requesting the data from the server requires some validation from the UPnP - User Profile Server in order to deliver the data to the User. The UP Server will contact the Control Point for authentication Check and data is provided. In our Existing System, the encryption process takes place between the UP Server and Control Point using AES and RSA Algorithm. Only one level of Encryption is used, i.e, applications developed shared in are subject to strict security review. The fundamental issue with respect to data is accuracy. Data Accuracy is not recognized.

3.2 PROPOSED SYSTEM:

In our Proposed System, we mainly focus to give the entire control to a common point called "Critical Control Point" which emphasis to discover the devices and list the devices. It has its entire access dependencies on a Profile Server, which will provide the UUID (Universal Unique Identifier) for accessing the network Data. The user is provided with a media for requesting the data from a Server. In-turn on receiving the request the server will contact the Critical Control Point to validate the User is authorized or not and data is Delivered to the respective User. With the purpose of offering the secured and accurate data to the User. Here, we use MD5 and AES algorithm in all the levels, i.e., from User via Server and Critical Control Point to User Profile Server.

4. SYSTEM DESIGN

System Design involves identification of classes their relationship as well as their collaboration. In objector, classes are divided into entity classes and control classes. The

Computer Aided Software Engineering (CASE) tools that are available commercially do not provide any assistance in this transition. CASE tools take advantage of Meta modeling that are helpful only after the construction of the class diagram. In the FUSION method some object-oriented approach likes Object Modeling Technique (OMT), Classes, and Responsibilities. Collaborators (CRC), etc, are used. Objector used the term"agents" to represent some of the hardware and software system. In Fusion method, there is no requirement phase, where a user will supply the initial requirement document. Any software project is worked out by both the analyst and the designer. The analyst creates the user case diagram. The designer creates the class diagram. But the designer can do this only after the analyst creates the use case diagram. Once the design is over, it is essential to decide which software is suitable for the application

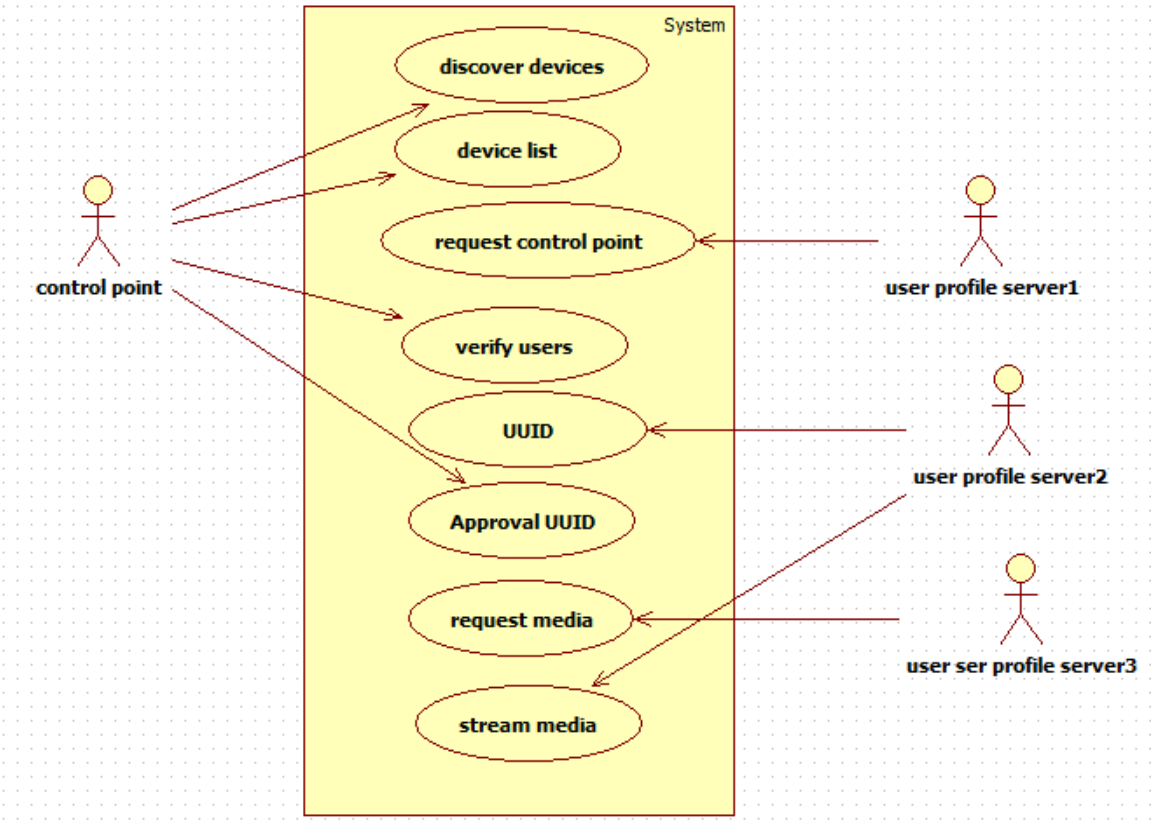
4.1 UML DIAGRAM OF THE PROJECT:

UML is a standard language for specifying, visualizing, and documenting of software systems and created by Object Management Group (OMG) in 1997.There are three important type of UML modeling are Structural model, Behavioral model, and Architecture model. To model a system the most important aspect is to capture the dynamic behavior which has some internal or external factors for making the interaction. These internal or external agents are known as actors. It consists of actors, use cases and their relationships. In this fig we represent the Use Case diagram for our project.

Use case Diagram:

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components a user or another system that will interact with the system modeled. A use case is an external view of the system that represents some action the user might perform in order to complete a task.

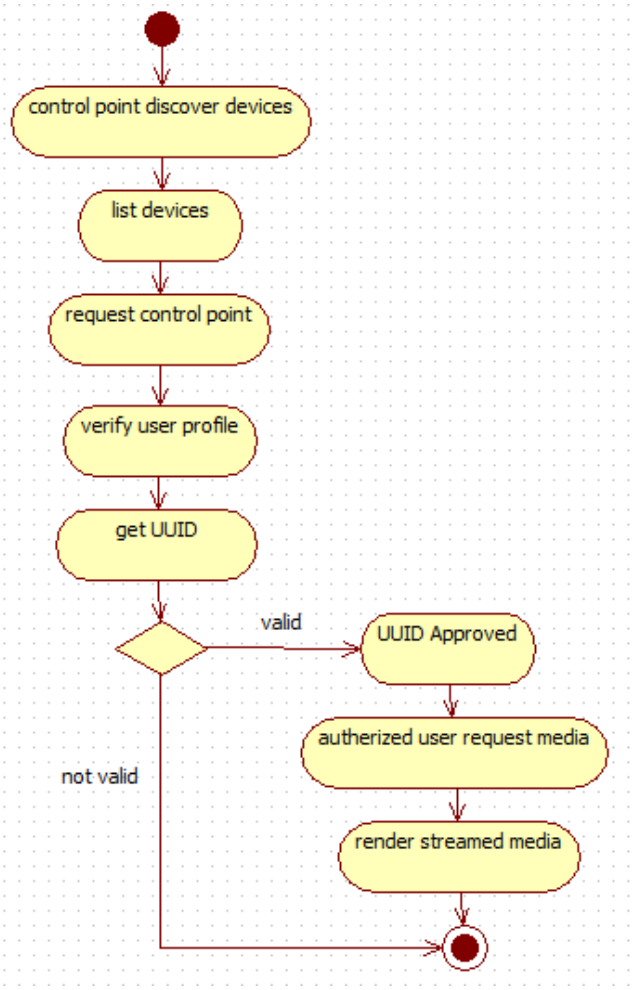
Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210



4.2 ACTIVITY DIAGRAM:

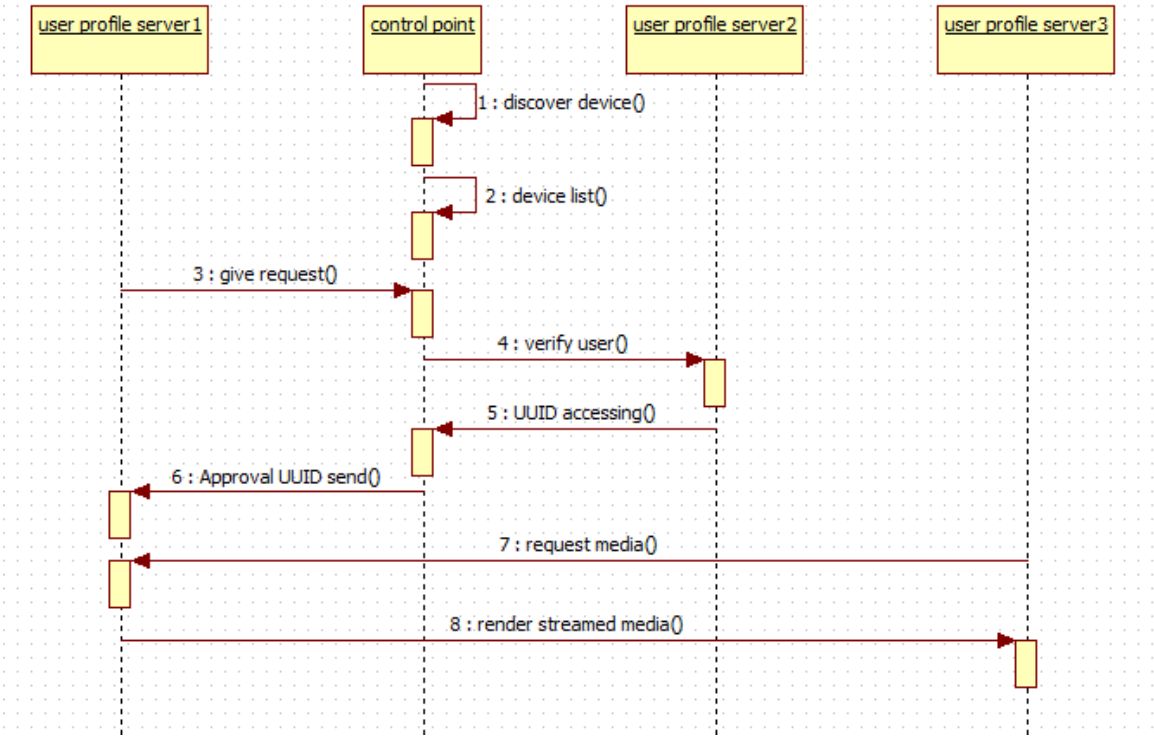
Activity diagram are typically used for business process modeling for modeling the logic captured by a single use case or usage scenario, or for modeling the detailed logic of a business rule. Although UML activity diagrams could potentially model the internal logic of a complex operation it would be far better to simply rewrite the operation so that it is simple enough that you don't requires an activity diagram. In many ways UML activity diagrams are the objects-oriented equivalent of flow charts and data flow diagrams (DFDs) from structured development.

Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210

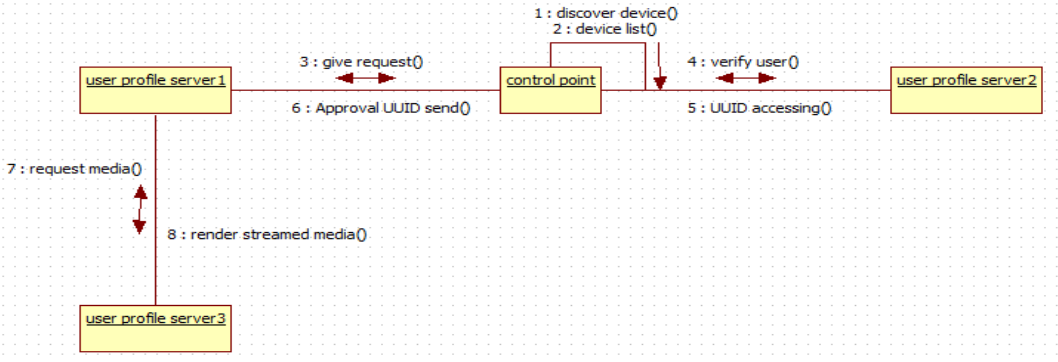


4.3 SEQUENCE DIAGRAM:

Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210



4.4 COLLOBORATION DIAGRAM:

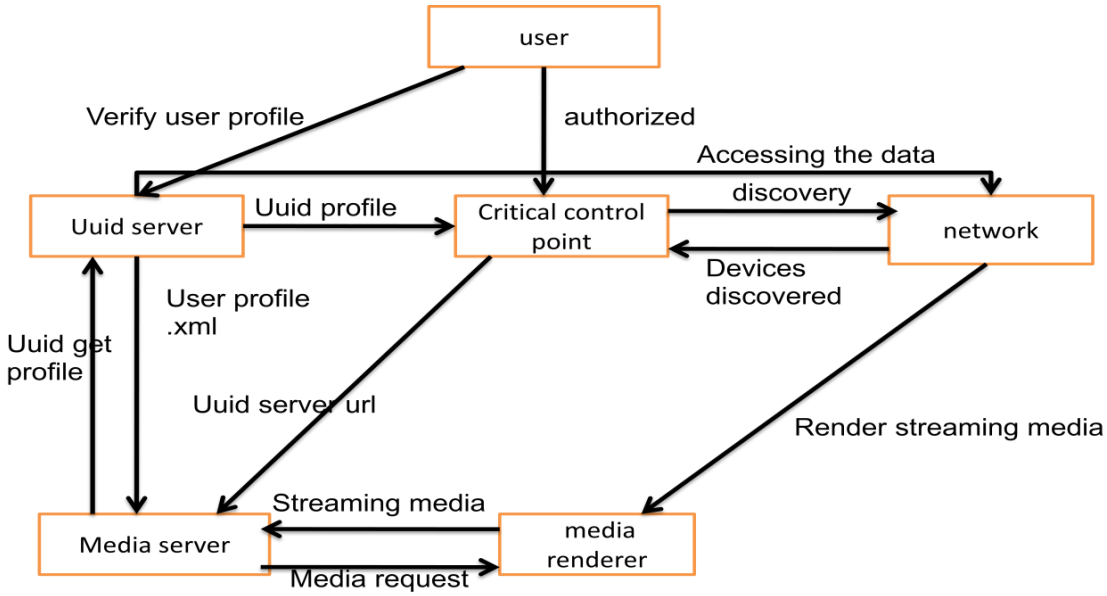


4.5 DATA FLOW DIAGRAM:

The Data Flow diagram is a graphic tool used for expressing system requirements in a graphical form. The DFD also known as the "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that to become program in system design.

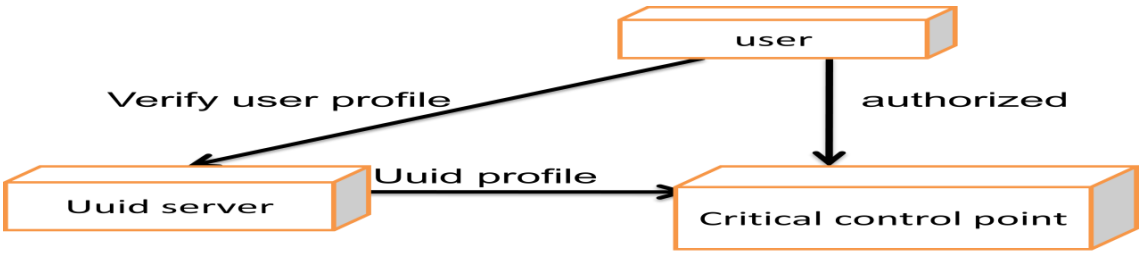
Thus DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail.

The DFD consist of series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system. A DFD describes what that data flow in rather than how they are processed. So it does not depend on hardware, software, data structure or file organization.

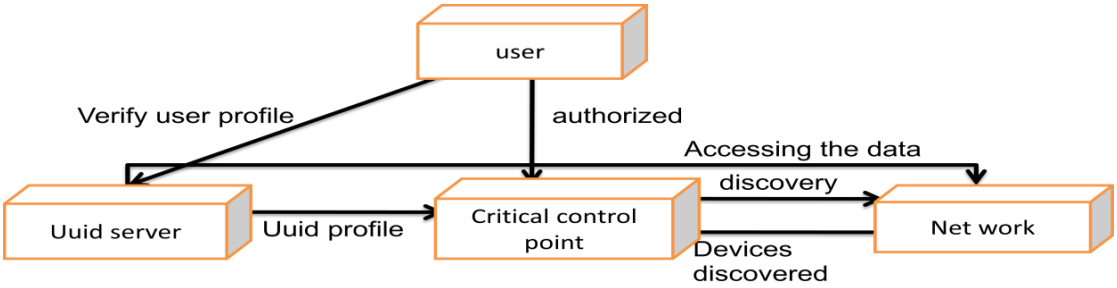


Over all dfd

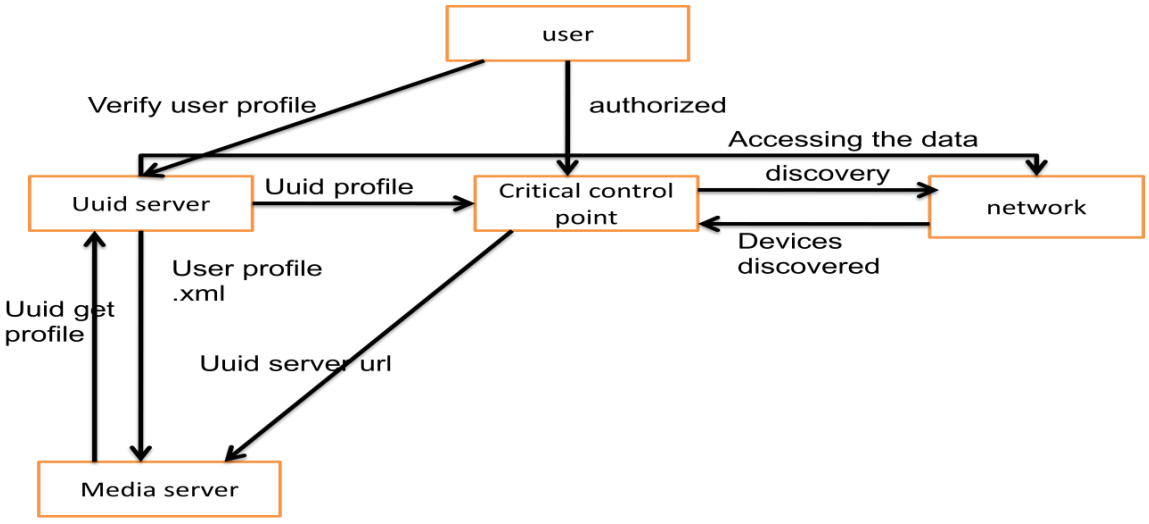
Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210



Dfd level 0:



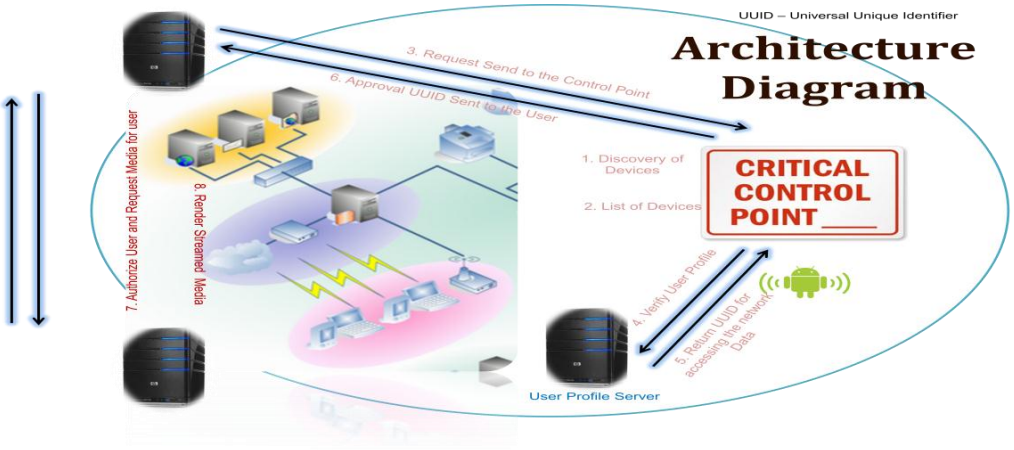
Dfd level 1:



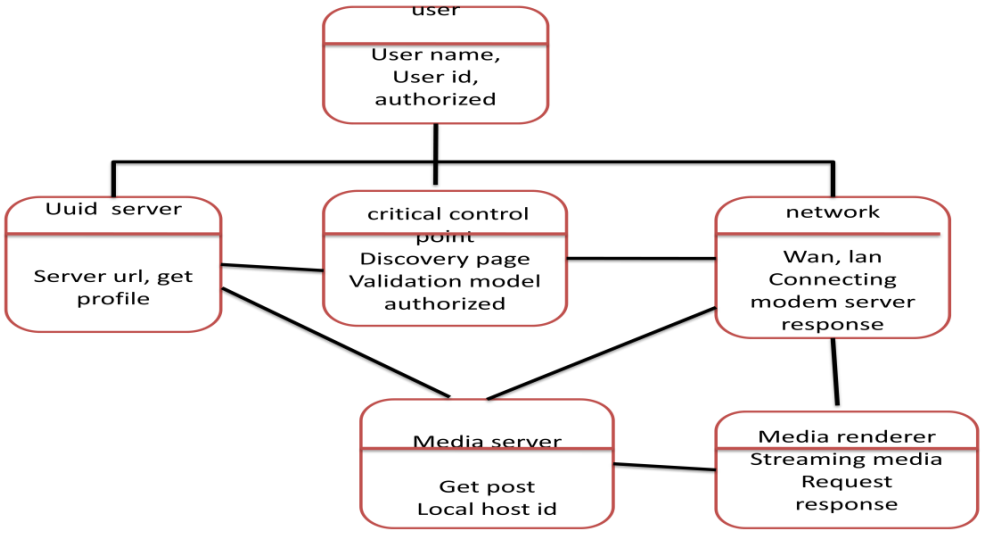
Dfd level 2:

Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210

4.6 ARCHITECTURAL DIAGRAM:

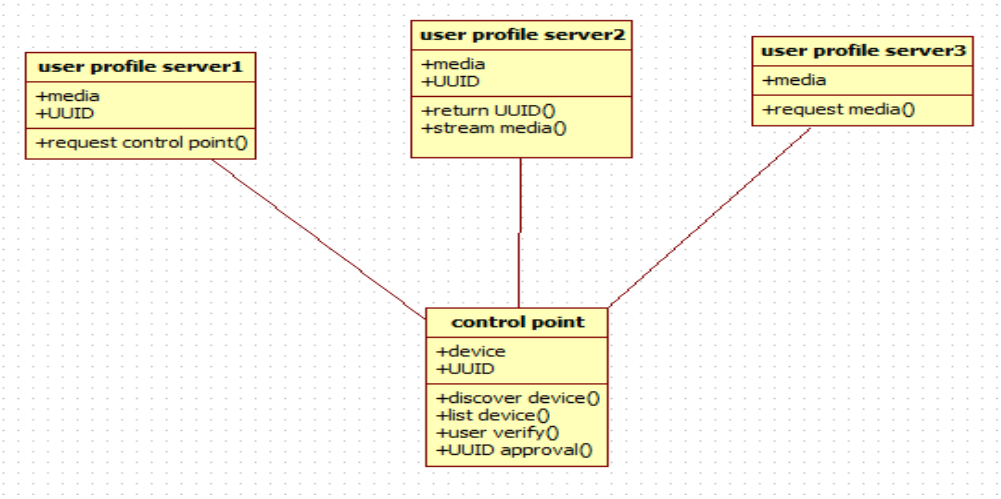


4.7 ER DIAGRAM:



Volume: 2 Issue: 2 25-Aug-2014,ISSN_NO: 2347 -7210

4.8 CLASS DIAGRAM



5. SYSTEM TESTING

5.1 TESTING OBJECTIVES

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

5.2 TYPES OF TESTS

5.2.1Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

5.2.2Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

5.2.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

5.3 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

6. CONCLUSION

In this paper, we mainly focused to give the entire control to a common point called "Critical Control Point" which emphasis to discover the devices and list the devices. It has its entire access dependencies on a Profile Server, which will provide the UUID (Universal Unique Identifier) for accessing the network Data

7. FUTURE ENHANCEMENT:

In the future enhancement the Critical Control Point will be able to discover the devices in a large range where the distance criteria plays a main role. With the appearance of the device the Critical Control Point may trace it out.

8. REFERENCES

[1] M. Jeronimo, J. Weast, UPnP Design by Example, Intel Press, Boston, 2003.
[2] H. Park, "Interoperability model for devices over heterogeneous home networks", IEEE Trans. Consumer Electronics, vol. 55, no. 3, pp. 1185 - 1191, Aug., 2009.

[3] J. Weast, “Advanced Universal Plug and Play Technology Topics”, IEEE Education & Learning, Jul., 2009

[4] V. Lortz, M. Saaranen, “DeviceProtection Service: 1”, Standardized DCP (SDCP), Version 1.0. UPnP Forum Committee, February, 2011.

[5] L. Kagal, T. Finin, and A. Joshi, “Trust-based security in pervasive computing environments”, Computer, vol. 34, pp. 154–157, Dec., 2001.

[6] L. Bauer, L. F. Cranor, M. K. Reiter, and K. Vaniea, “Lessons learned from the deployment of a smartphone-based access-control system”,
Proc. of the 3rd Symposium on Usable Privacy and Security, SOUPS ’07. New York, NY, USA: ACM, 2007, pp. 64–75.

[7] J. Kim, Z. Kim, and K. Kim, “A lightweight privacy preserving authentication and access control scheme for ubiquitous computing environment”, Proc. of The 10th International Conference on Information Security and Cryptology, Berlin, Heidelberg: Springer- Verlag, 2007, pp. 37–48. 152 IEEE Transactions on Consumer Electronics, Vol. 59, No. 1, February 2013

[8] J. Yves Tigli, S. Lavirotte, G. Rey, V. Hourdin, and M. Riveill, “Context-aware authorization in highly dynamic environments”, IJCSI International Journal of Computer Science Issues, vol 4, no. 1, pp. 1694-0784, Nov., 2009.

[9] M. L. Damiani and C. Silvestri, “Towards movement-aware access control”, Proc. of the ACM GIS International Workshop on Security and Privacy in GIS and LBS, New York, NY, USA: ACM, 2008, pp. 39–45.

[10] H. Nakashima, H. Aghajan, and J. C. Augusto, Handbook of Ambient Intelligence and Smart Environments, 1st ed. Springer Publishing Company, Incorporated, 2009.

[11] A. Klemets and B. Da Costa, Upnp Authentication and Authorization Patent, U.S. Patent 7 882 356, 2011.

[12] J. Karl, Upnp CDS USER PROFILE, Rancho Santa Margarita, CA. U.S. Patent 8 185 949, 2012.

[13] Zhu, X.; Shi, Y.-Y.; Kim, H.-G.; Eom, K.-W.; , "An integrated music recommendation system", IEEE Transaction on Consumer Electronics, vol.52, no.3, pp.917-925, August, 2006.

[14] Bhaskar C,Rajapirian P," Precision Aware Self-Quantizing Hardware Architecture for the Discrete Wavelet Transform.",vol: 2-3 (Pages 98-104), *ISR Journal*, Available:http://isrjournals.org/archives_abstract.php?id=28&t_n=ijarcseit&d_id=66&dm

[15]R.Jegadeesan ,Dr.N.Sankar Ram,M.Naveen Kumar," Less Cost Any Routing With Energy Cost Optimization",vol: 2-1 (Pages 85-90), *ISR Journal*, Available:http://isrjournals.org/archives_abstract.php?id=28&t_n=ijarcseit&d_id=66&dm