Virtual Device: Media Service Fitness, Selection and Composition

Niall Murray, Brian Lee, A.K. Karunakar, Yuansong Qiao, and Enda Fallon

Software Research Institute,
Athlone Institute of Technology, Ireland
{nmurray,ysqiao,akkarunakar}@research.ait.ie,
{blee,efallon}@ait.ie

Abstract. The motivation for the virtual device is to facilitate the seamless use of application services residing on different devices in the vicinity of the user. Due to heterogeneity of devices in the user's pervasive environment, multiple potential combinations to support a required task may exist. This work aims to determine the best possible media services considering all instances. The best services are selected and composed to satisfy a user task. Recent works propose using user preferences, environment capabilities and similarity between requested and available services to determine service fitness. We consider potential local and remote content sources and destination devices, with similarity, user preferences and environment capabilities to determine fitness. Services are selected for composition based on fitness. We model and simulate this issue and explain the results of our experimentation. Optimal multimedia service composition from varying devices provides the user with the best possible multimedia consumption and production experience.

Keywords: Virtual device, Atomic service fitness, service composition, media service selection.

1 Introduction

Future multimedia networks will be a ubiquitous communication platform where users will enjoy continuous multimedia services in any location on any device [1]. Today users are surrounded by technology that is heterogeneous, pervasive, and variable [2]. Consequently a user has a rich set of media services available to them when undertaking multimedia tasks in a multimodal, device independent manner. The virtual device supports user tasks such as audio video communication by combining media services from different devices. These devices include; small hand held multi-functional devices with limited processing and display capabilities, enhanced single function dedicated devices or powerful multifunctional multimedia systems (e.g. PDA, PCs, HDTV's, Network Speakers, surround sound systems).

Considering these devices, overlaps may exist in the types of services provided. The fitness (suitability) of individual services is calculated to distinguish between instances. By selecting and composing the blue-chip services of different devices, the

N. Venkatasubramanian et al. (Eds.): Mobilware 2011, LNICST 93, pp. 181–194, 2012.

virtual device supports user task satisfaction beyond what a user companion device can offer. The principles of Service Oriented Architecture (SOA) provide the basis for a suitable approach. The focus of this paper is one of those principles, service composition [3].

A service is defined as an indivisible, self contained application unit that performs a processing function on multimedia content. Internet accessible consumer devices provide one-to-many services. The virtual device supports the user task by composing the best service(s) of different devices within the context of the same session. The "single service for every device" vision of the webinos project [4] fits neatly into this paradigm.

Fig. 1 below illustrates the types of services we refer to. It also includes what we define as content manipulation services, which is outside the scope of this work.

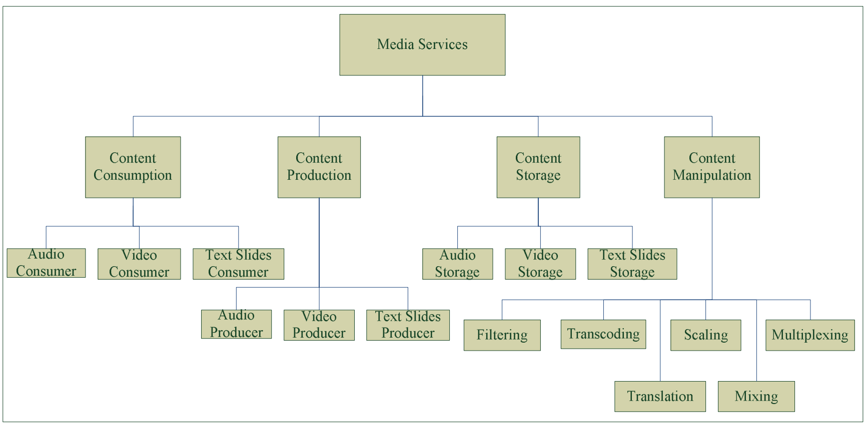

Fig. 1. Multimedia Content Service types

Concurrent services require strict functional and/or timely synchronization [5]. It is a multi-source, multi-destination service composition challenge. A natural method on which to model task or composite multimedia service composition is using graph theory as has been done in [6][7][8][9][10].

This work focuses on how to achieve the maximum quality media service composition by considering context. If this selection process is not performed properly, the search will generate non optimized results, causing a low Quality of Service (QoS) perception from the consumer point of view [11]. Simply choosing the most powerful device or service does not always result in the most efficient and most favourable user experience. It may not be the user's preferred mode of communication. In addition, there is no advantage in choosing a powerful display device if a very limited video production device is producing the video content. Unnecessary execution of manipulation content services in Fig. 1 can be minimized through intelligent decision making.

We propose the consideration of local and remote content consumption and production services in selecting local services, in conjunction with user preferences and device resources. Considering remote services enables early elimination of unusable services and we suggest that it facilitates reduced media processing costs and delays i.e. if we can match a particular encoding format source and destination between distributed compositions – no transcoding service would be required. Finally, it provides a method to overcome the dependency [12] on service users' feedback.

The overall contributions of this paper are:

1. Defining the service selection and composition problem to include local and remote services as well as user preferences and device resources.
2. We model and define the fitness of a service as a function of device resources, user preferences, availability, encodings and potential remote services.
3. We present the results of our experiments displaying the benefits of using our proposed algorithm.

This paper is organized as follows: Section 2 defines our problem with the aid of a sample scenario. Section 3 describes how the service composition problem is modelled in terms of local services, user preferences, device resources and remote services. Section 4 presents the algorithms for service composition. Section 5 explains our experimentation and simulation results. Section 6 outlines related work. Finally in section 7 we conclude this paper, discuss our contributions and outline future work.

2 Problem Definition

We envisage public and private environments consisting of multiple mobile and stationary nodes. These nodes provide one or more content related services as per Fig. 2a, which shows an audio-video camera providing five possible services. All nodes are Internet connected devices where resident services can be invoked by peer nodes. These devices may include; small hand held multi-functional devices with limited processing and display capabilities, enhanced single function dedicated devices or multifunctional multimedia systems.

Consider the following scenario defined as real time communication between distributed compositions. Brian is talking with John on his virtual device enabled smart phone and is walking from his office to his car. John is working from home and is using the microphone from his personal computer and surround sound speaker system to converse with Brian. Also in John's study are a PC connected camera and a large display screen. Once Brian sits in his car and places the phone in the holder, his call is automatically transferred from his handheld device to the services available in his car. He is now using the microphone embedded in the sun visor, speakers from the car radio system and can see John on the LCD panel integrated into the car panel. John can also now see Brian because of the camera built into the steering wheel. Local services from Brian's perspective are the services resident on devices within his vicinity that he uses to communicate with John (e.g. the microphone embedded in the visor). Remote services from Brian's perspective are the services available to John for communication (e.g. microphone in his personal computer).

Considering this scenario and Fig. 2b, many of the devices in a user's vicinity may provide similar functional services. Differences may exist between these services in terms of capabilities, availabilities, user preferences and usage costs. In the scenario outlined, the user may want to use a number of different types of services (e.g. video and audio consumption and production services) in a communication session. Consequently there are multiple constraints in terms of what the user requires, and multiple choices in terms of devices providing different services to solve these problems. Like [2], we consider determining the quality of compositions a variant of a 0-1 Knapsack problem, named multiple dimensional, multiple choice 0-1 Knapsack. The multiple dimensions refer to the multiple constraints and the multiple choices refer to choosing one among a set of similar items. In optimizations research this problem has been proven to be NP-complete [13].

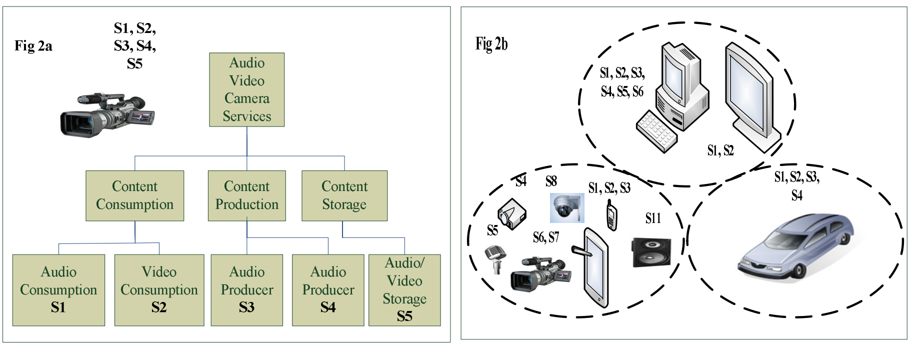

Fig. 2. (a) shows how a video camera device can provide many content related services. **(b)** shows a flavour of some of the devices providing one-to-many services denoted by S_i that could be used in service composition to create the virtual device.

Our work aims to achieve the best service composition by selecting the highest scoring instances of the individual service types, through consideration of local and remote services, device resources and user preferences.

3 Service Composition System

To compose services from distributed devices, a number of steps are required. Many of these steps are outside the scope of this document but for completeness we air some views on potential approaches. As per Fig. 3 below, it is assumed that an Internet scale network exists. Devices provide their content production, storage, consumption and manipulation services for execution by peers. The following entities are the basis for the system.

- *User Companion device*: it is assumed that a task requiring atomic service composition is initiated from the user companion device. It is likely that the user companion device will always be in the vicinity of the user and hence, in the vicinity of devices providing atomic services.
- *Atomic service*: an application level service that provides content processing functionality in one of the four defined areas of consumption, production, storage and manipulation.

- *Service component devices*: mobile or stationary devices hosting any of the four types of content services of Fig. 1. The user companion device can also provide atomic services for composition.
- *Coordinator*: it coordinates the discovery and selection of the atomic services. It is typically a mobile or stationary device that has high computation and processing ability and is in the vicinity of devices supporting a rich set of atomic services. The selection of the coordinator device has been addressed in [14][15]. The coordinator executes the algorithm introduced as part of this work.
- *Composite service*: multiple atomic services composed to satisfy a user task.
- *Task*: an application level user request that is satisfied by an atomic service or composite service.
- *Service composition specification*: is a specification of a user defined task. Considering the sample scenario in section 3, it is a specification of the atomic services required for real time communication between distributed compositions.

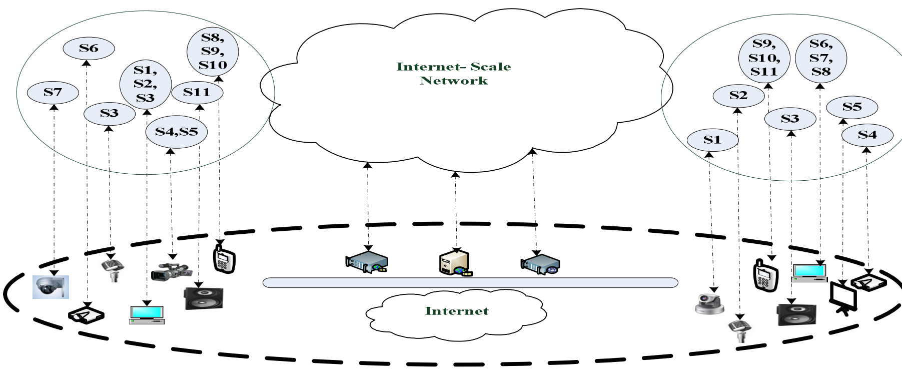

Fig. 3. Internet-Scale Network where devices provide their services for composition to support communication between distributed compositions

3.1 Network Model

This network can be represented by a graph, *G*. Each node *N*, in the graph represents a device that provides one-to-many services. *F* represents a set of links between each of the nodes.

$$G = \{N, F\} \tag{1}$$

The resources of a device reflect the non functional characteristics. The resources *R(n)* are quantifiable [16] values such as CPU (*X*), battery power (*P*), network bandwidth (*B*) and memory *(M)*. These resources are used to calculate the fitness of the service component device that supports the service. This also provides a measure of service availability. The value of *R(n)* for each device is a function of local device values X_L, P_L ,B_L and $\mathrm{M_L}$ and the maximum values of X_{max}, P_{max}, B_{max} and $\mathrm{M_{max}}$ considering all devices that provide services.

$$R(n) = f(X, P, B, M) \tag{2}$$

$$R(n) = \left(X_L / X_{\max} + P_L / P_{\max} + B_L / B_{\max} + M_L / M_{\max} \right) \div 4 \text{ ,where } 0 \le \mathrm{R(n)} \le 1$$

The services available on a particular node S = {S_1, S_2,…S_k} is a set of Fig. 1 type services. The services supported by a device *n* are denoted by $S_i(n)$ where *ST(s)*

$$S_i(n) = \{ST, E, A\} \quad \text{, where } 0 \le \mathrm{i} \le \mathrm{k} \tag{3}$$

represents the service type as per Fig. 1, *E* represents the content compression formats supported and $A(s_i)$ represents the availability status of the service. It is assumed that all nodes in the network can exchange information.

3.2 Service Composition Specification

In order to compose a set of services to address a user task, a specification of the services required is necessary. Similar to [6][7][9], we detail this specification using graph theory. Like in [17], a multi level approach to service composition is employed. Local and remote services, availability, device resources and user preference are considered. The user's task is modelled as a task graph, TG.

$$TG = \{S_L, M_i, S_r, U, F\} \tag{4}$$

$$S_L \subset (C_L \cup P_L) and S_r \subset (C_r \cup P_r)$$

S_L are the local content consumption and production services. S_r are the remote content consumption and production services. C_L and P_L are the local content consumption and production services respectively whilst C_r and P_r are the remote content consumption and production services respectively. M_i represents content manipulation services. *F* represents the data flows between P_r to C_L and P_L to C_r potentially through M_i intermediary manipulation services. *U* is a set of user preferences related to consumption and production services for the task. User preferences are specified as part of the task description and explicitly provided by the user once available services are determined. The novel aspect of this work is the use of P_r and C_r to decide the combination of P_L and C_L selected. It is assumed that nodes providing services for composition are physically located close to each other.

Real time communication between distributed compositions will result in multiple concurrent bi-directional content streams. The aim is to select a set of local services, mapping from *TG* to *G*, considering user preferences, device resources, local and remote services to achieve the best context based composition.

For all nodes *N*, multiple services may exist providing similar functionality. Selection of different instances of the same service types, results in different mappings from the service specification to device's services. Between services, different values exist in how closely a service instance matches a service request. A user may prefer one service instance over another. Different values of *R(n)* will exist between devices that host the same service types. Considering these factors, the goal is to select the best possible composition set for the user.

3.3 Local Atomic Service Fitness Calculation Considering Local Information

A two step suitability function is used to calculate the fitness of atomic services of the same type. We consider the QoS function of Perttunen et al. [18] an excellent base to perform this task. This model is extended to produce atomic service type league tables based on atomic service fitness. We assume a reasonably small number of atomic services for each service type are discovered; otherwise per service fitness calculation could become an exhaustive process. Equation 5 shows each of the parameters used to generate atomic service fitness. The maximum value for each of these parameters is one; hence highest potential atomic service value (ASV) is one. A league table is created based on the ASV for each of the services (i) within each atomic service type (t).

$$ASV_t(s_i) = R(n) \times W_t(s) \times Sim(RS_i, AS_i) \times U(s_i) \times A(s_i) \qquad (5)$$

- *R(n)* as previously introduced denotes the device resources of the service component device hosting the service.
- $W_t(s)$ [18] is a user inputted weight associated with a particular service type specified by the user. It signifies how important a user views one particular service type e.g. the user may feel audio is a more important service to have over video in a given context, hence a higher weighting is selected for audio.
- $Sim(RS_i, AS_i)$ [18] is a similarity function comparing the requested service (RS_i,) and available service (AS_i). Certain user preferences like preferred screen size are considered as part of the similarity function execution. This parameter is not applicable for all service types. If it is not applicable, a default value of one is applied.
- $A(s_i)$ represents if an atomic service is available for use as part of a composite service. In [18], the availability function is a measure of current availability state, usage policy, lease information, queue for usage and user groups. For simplicity, only current availability state is considered here.
- $U(s_i)$ is the user preference for a particular atomic service instance. The incorporated user preference model is derived from [16]. Users specify a preference value reflecting their approval or disapproval for a particular service instance. An option of "Avoid this service instance" facilitates the user to specify that they do not wish to instantiate a particular service instance.

Table 1. User Preferences Table

Importance Level	Preference Value
Desired Service Instance	1.0
Satisfactory Service Instance	0.7
Acceptable Service Instance	0.3
Avoid this service Instance	0.0

3.4 Local Atomic Service Fitness Calculation Considering Remote Services

The next contribution of this paper involves a comparison of all atomic service tables with the equivalent remote services. The benefits of this step are: (1) it ensures all required services can be supported i.e. what the user would like to do is actually possible, considering real time communication between distributed compositions. (2) It checks for commonalities in terms of encoding formats between the potential local and remote content sources and destinations. Considering this removes the necessity for execution of some of the intermediary manipulation services (e.g. transcoding), thus reducing processing costs and delays. For local services that have commonalities with a remote side, we add an additional weight to the $ASV_t(s_i)$ which results in the final service value ($FSV_t(s_i)$). Local sources that have a remote "partner" service and support a common encoding are given a default value of 1 for use in service fitness calculation. Local sources that have a remote "partner" service but do support a common encoding format are given a default value of 0.5 for use in service fitness calculation. Finally, local services that do not have a remote "partner" service are given a default value of 0 for use in service fitness calculation. If a partner service does not exist, this means that this service cannot be used in the communications session; hence the value of 0 is assigned. As shown in table 2, $FSV_t(s_i)$ is the product of the respective weightings and $ASV_t(s_i)$.

3.5 Atomic Service Composition

The final league tables comprise atomic services ordered in terms of $FSV_t(s_i)$. These tables represent the set of all available atomic services for the communication session. The top scoring services in each table are selected for composition. These services make the selected service set (SSS). If during the session, a particular service becomes unavailable (for example due to node mobility), the next most suitable service type can be easily retrieved from the respective table. Hence this system provides an efficient service failure recovery mechanism. Only one service is selected from each of the service tables at any one time. The resultant selected composition is a set of the highest scoring services calculated based of $FSV_t(s_i)$. The overall rating for this composition is the sum of the max $FSV_t(s_i)$ for each atomic service type.

4 Service Fitness and Composition Algorithms

This section outlines in pseudo code the various steps of the two algorithms. Service discovery provides the coordinator with information regarding available atomic services to satisfy the user task. This information is used to calculate the service availability $A(s_i)$ and device resource capabilities $R(n)$. It also provides the available service information (AS_i) required for $Sim(RS_i, AS_i)$. The user inputs their user preferences $U(s_i)$, weightings for service type $W_t(s)$ and the requested service information (RS_i). Considering these inputs from the discovery and user, it is possible to execute Algorithm 1 and calculate $ASV_t(s_i)$ as per equation 5.

Algorithm 1. Generation of Atomic Service Value ($ASV_t(s_i)$)Tables

```
List of required atomic services (RAS) to satisfy task
List of local discovered atomic services (DAS) to satisfy task
Create a fitness table for each required atomic service type
For each RAS type
    For each DAS type, If match( RAS, DAS) // service types match
            Determine Sim(RS_i, AS_i)// similarity
            Determine A(s_i) // service availability
             Determine R(n) //resource capability from [X,P,B,M]
            Determine U(s_i) // user preference
            Determine W_t(s) // atomic service weighting
        Calculate ASV_t(s_i)
        Upload score & details to fitness table
Return FitnessTable
```

Additional differences may exist between atomic service types in terms of compression formats supported. This information is used to further calculate the fitness of the local atomic services considering the formats of the remote services. Algorithm 2 compares the services in the tables generated in Algorithm 1 with the remote services. The aim is to find remote "partner" services for each of the local atomic service types e.g. a local video consumption device could partner with one-to-many remote video production services. If no "partner" service is found, the score for this service instance is set to 0 as it is not possible to support this service type in the communication session. If a partner service is found, a comparison in terms of encoding formats supported is executed. If a common encoding exists, an extra weight is added to this particular local service as per lines 6 and 7 of Algorithm 2. Table 2, column "Algorithm 2 (Encoding Match)" shows that encoding matches were found for the following services: HDTV, panel display, smart phone display and hence the weighting value of one. No encoding match was found for the PC Display and as a result it is given the weighting of 0.5. The highest scoring atomic services in each of the final set of fitness tables are selected for composition as discussed. The composition is the highest quality mapping from *TG* to *G*.

Algorithm 2. Comparison of Atomic Services with Remote Services and Optimal Service Composition

```
1.  Local atomic service fitness tables to satisfy task
2.  Remote atomic services
3.  For each Service Type table
4.      For each Remote AS, check service type //e.g. video consumer
5.          If partner(DAS, RAS) // service types
6.              If (encodings match) //
7.              Update FSV_t(s_i) // see table 2
8.      return All Fitness Tables
14.     For each FSV table
15.         SelectTopScoringService()
9.  Compose Set of setOfTopScoringServices
```

5 Simulation Results and Analysis

Simulations are performed on a Windows Vista Ultimate OS with 4.00 GB RAM and Intel® Core™ 2 Quad Q6600 @ 2.4GHz. The simulated environment models ten devices that can potentially provide one-to-many services within the vicinity of the user. Table 2 below shows a sample generated league table for a video consumption atomic service showing all execution points of Algorithms 1 and 2. As mentioned previously, the maximum value for each of the parameters is one, hence the highest possible scores of $ASV_t(s_i)$ and $FSV_t(s_i)$ is also one. Inspecting the values, considering resource capability ($R(n)$) and similarity with requested service $Sim(RS_i, AS_i)$, the PC Display service scores as the strongest candidate. All instances are available and given the same weighting, hence $A(s_i)$ and $W_t(s)$ are equal to one. It is not until Algorithm 2 and encodings matching takes place that the HDTV service scores highest and rises to first position in the table with the PC Display moving to second place. Based on this execution, the HDTV video consumer service is selected as part of the composition to satisfy the user task.

Table 2. Video Consumption $FSV_t(s_i)$ Fitness table after execution of Algorithm 1 and 2

AS Type: Video Consumer	R(n)	$Sim(RS_i, AS_i)$	$A(s_i)$	$U(s_i)$	$W_t(s)$	$ASV_t(s_i)$	Algorithm 2 (Encoding Match)	$FSV_t(s_i)$
HDTV	0.93	0.857	1.0	0.7	1.0	0.56	1.0	0.56
PC Display	1.0	0.928	1.0	0.7	1.0	0.65	0.5	0.325
Panel Display	0.85	0.5	1.0	0.3	1.0	0.127	1.0	0.127
Phone Display	0.73	0.333	1.0	0.3	1.0	0.073	1.0	0.073

Figure 4a below reflects the ratings of the services as their information is processed through each of the stages of Algorithms 1 and 2. $R(n)$ reflects the resource capabilities of the devices supporting the services. The $Sim(RS_i,AS_i)$ function determines the similarity between the user request and the available services. $U(s_i)/ A(s_i)/W_t(s)$ are the user preferences, availability and weightings for each of the services. Once these five parameters are executed, $ASV_t(s_i)$ is calculated which reflects Algorithm 1 processing. Algorithm 2 performs the encoding matching check which as illustrated results in a changing of places between the HDTV and PC Display services. Because a common codec exists between the HDTV and remote video production service, an extra weighting is given to the HDTV display service which results in it becoming the highest scoring service in the video consumer service table.

As mentioned earlier, the composed services are a set of the highest individual scoring atomic services. For explanation purposes, Fig. 4b compares the fitness of the highest scoring SSS with a set comprised of medium scoring atomic services, and two compositions with sets of low scoring atomic services. Comparing the highest SSS with services (s1,s6,s10,s14) and lowest SSS with services (s4,s8,s12,s17) have scores of 3.2 and 0.7 respectively as shown in Fig. 4b. This clearly demonstrates the validity of the algorithms and the approach taken as part of this work.

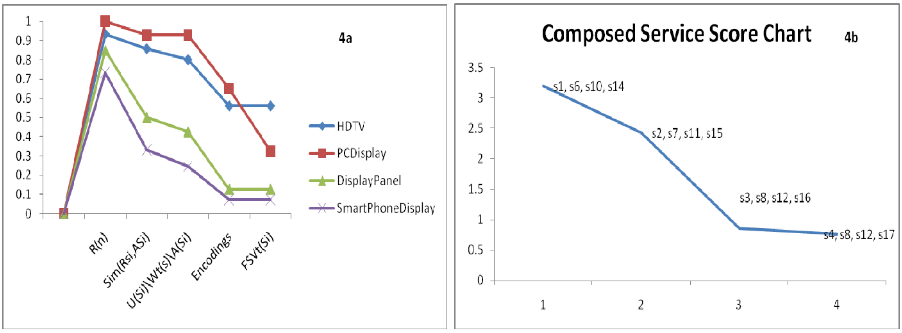

Fig. 4. 4a reflects the change in scores after each of the steps of equation 4. **4b** compares the scores of a set of high fitness services with a set of medium fitness services and a set of low fitness services calculated. One atomic service from each of: Audio Consumption set (s1,s2,s3,s4,s5), Video Consumption set (s6,s7,s8,s9), Audio Production set (s10, s11, s12, s13) and Video Production set (s14, s15, s16, s17) are selected for each composition.

6 Related Work

A broad range of related work involving task based service composition in pervasive environments and MANETS, content delivery and adaptation, the connection of devices in pervasive environments exists.

Similar to our work, Sousa et al., in [2] describe an approach to finding the best match between the user's requirements and the environments capabilities. Hossain et al., in [19] determine the best possible composition as a function of gain (the extent of which a media service satisfies a user in a particular context) and cost of the service. In [20], Karmouch et al. define service composition in MANETS as a distributed constraint satisfaction problem. Similar to us they use a QoS model based on the work of Pertuttan et al., [18]. We borrow facets of and extend this model with device capability and consideration of the remote services. Mukhtar et al., define an approach for task composition considering user preferences and device capabilities [16]. In [9], they use graph theory to define the user task. In [12] Atrey et al., use how regularly a service has been composed with another service to determine a reputation for a service. In [21] Jiang et al., address service composition based on prospect of minimum disruption. None of these works consider the potential atomic remote services to select local atomic services.

In relation to selection of the coordinator device, Karmouch et al., in [15] implement a broker based distributed service composition protocol which extends the work of Chakraborty et al., [14] for service composition. Basu et al., [7] define graph based approaches to distributed application composition approaches in MANETS.

[22][23][24] propose different approaches for connecting devices in pervasive environments. In [22], Schuster et al., provide a service orientated architecture for virtual device composition utilizing middleware on all devices. In [23], Senthivel et al., construct ad hoc service overlay networks (SON) based on service requests. In [24], Park et al., propose an interoperability framework based on the JXTA protocol. In [25], Ibrahim et al., survey middleware approaches to service composition and define service composition as a four step process: translation, generation, evaluation and execution. In [10], Kalasapur et al., propose a SOA based middleware platform which also incorporates graph theory. In [26], Lee et al., propose an approach based on a virtual device software manager, a middleware manager and hardware adaptation. In [27], Grigoras et al., address MANET formation based on device constraints like bandwidth and battery power.

Much service composition research has focused on media delivery. In [28], Gu et al., propose SpiderNet which provides Statistical QoS assurances for service composition. In [29], Jafarpour et al., strategically place content adaptation nodes in an overlay network to reduce costs in terms of communication and computation. In [30], Qian et al., determine lowest delay in service oriented multimedia delivery in pervasive environments. Xu et al., [31] propose a distributed Storage-assisted Data-driven overlay network to support P2P Video-on-Demand services.

In [32] Nahrstedt et al., introduce and discuss challenges with web services based approaches to multimedia delivery. SPovNet [33] is an overlay based solution that facilitates the spontaneous deployment of distributed network applications and services. In [34], Kim et al., discuss an emerging trend of media orientated service composition with SON's and outline challenges. They also dicuss virtualized resource components as a futuristic solution. In [35], Buford et al., suggest an Internet-scale P2P Overlay to facilitate expanding the capability of a device.

To the best of our knowledge, the use of remote potential services has not been a driver to calculate fitness, select and compose local atomic services. In [6], Nahrstedt et al., discuss a novel scoring function based on importance of objects in media streams to decide what is displayed on what portion of a shared screen. In our work, it is the capability and the availability of services in the remote composition that we use as an eliminator for irrelevant service components as part of the composition process.

7 Conclusion

This paper has presented a framework to calculate service fitness and select services for composition. This novel approach uses the remote potential capabilities as input to deciding the local service composition in addition to user preferences and resource capabilities. We formally model the user task taking into consideration all of the aforementioned factors. A two step suitability algorithm produces league tables for required atomic services based on service fitness. The result provides a user with an optimized selection of services, whilst providing an efficient service failure recovery mechanism. The simulation and experimentation show how each of the factors considered; local services, remote services, resource capability and user preferences affect the service scoring and how the best possible set of services are selected. In future work, we aim to define group synchronization schemes to address requirements of cluster applications where multiple correlated media stream with variable end-to-end delays exist.

Acknowledgments. We would like to recognize the assistance of Enterprise Ireland through its Applied Research Enhancements program in the financing of this Research. Also Dr. Mark Daly, a lecturer in the AIT School of Engineering for his assistance and advice with system modelling.

References

1. Murray, N., Qiao, Y., Lee, B., Karunakar, A.K., Fallon, E.: Design Considerations for Future Multimedia Systems: A Comparison of Approaches Based on SIP and the Advanced Multimedia System. International Journal of Ambient Computing and Intelligence 3(1), 20–32 (2011)
2. Sousa, J.P., Poladian, V., Garlan, D., Schmerl, B., Shaw, M.: Task-based Adaptation for Ubiquitous Computing. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews 36, 328–340 (2006)
3. Erl, T.: SOA Principles, `http://www.soaprinciples.com/`
4. webinos (Secure Web Operating System Application Delivery Environment), FP7-ICT-2009-5 – Objective 1.2, `http://webinos.org/`
5. Nahrstedt, K., Balke, W.-T.: A Taxonomy for Multimedia Service Composition. In: 12th ACM International Conference on Multimedia (2004)
6. Nahrstedt, K., Yu, B., Liang, J., Cui, Y.: Hourglass multimedia content and service composition framework for smart room environments. Journal on Pervasive and Mobile Computing 1(1), 43–75 (2005)
7. Basu, P., Wang, K., Little, T.D.C.: Dynamic Task Based Anycasting in Mobile Ad Hoc Networks. Journal Mobile Networks and Applications (2003)
8. Gu, X., Nahrstedt, K.: On Composing Stream Applications in Peer-to-Peer Environments. IEEE Trans. on Parallel and Distributed Systems 17(8), 824–837 (2005)
9. Mukhtar, H., Belaïd, D., Bernard, G.: A Graph-Based Approach for Ad hoc Task Composition Considering User Preferences and Device Capabilities. In: GLOBECOM Workshops. Current version (2009)
10. Kalasapur, S., Kumar, M., Shirazi, B.A.: Dynamic Service Composition in Pervasive Computing. IEEE Transactions on Parallel and Distributed Systems 18(7) (2007)
11. Dutra, R.G., Martucci Jr., M.: Dynamic Adaptive Middleware Services for Service Selection in Mobile Ad-Hoc Networks. In: Cai, Y., Magedanz, T., Li, M., Xia, J., Giannelli, C. (eds.) Mobilware 2010, Part IV. LNICST, vol. 48, pp. 189–202. Springer, Heidelberg (2010)
12. Atrey, P.K., Hossain, M.A., El Saddik, A.: A Method for Computing the Reputation of in Multimedia Services through Selection and Composition. Journal Mobile Networks and Applications 13(6) (2008)
13. Pisinger, D.: An Exact algorithm for Large Multiple Knapsack Problems. European Journal of Operational Research 114 (1999)
14. Chakraborty, D., Joshi, A., Finin, T., Yesha, Y.: Service Composition for Mobile Environments. Mobile Network and Applications 10(4), 435–451 (2005)
15. Karmouch, E., Nayak, A.: A Distributed Protocol for Virtual Device Composition in Mobile Ad Hoc Networks. In: IEEE International Conference on Communications (2009)

16. Mukhtar, H., Belaïd, D., Bernard, G.: User Preferences-Based Automatic Device Selection for Multimedia User Tasks in Pervasive Environments. In: 5th Conference on Networking and Services (2009)
17. Zhang, B., Shi, Y., Xiao, X.: A Policy-Driven Service Composition Method for Adaptation in Pervasive Computing Environment. The Computer Journal 53(2), 152–165 (2007)
18. Perttunen, M., Jurmu, M., Riekki, J.: A QoS Model for Task-Based Service Composition. In: Workshop on Managing Ubiquitous Communications and Services (2007)
19. Hossain, M.A., Atrey, P.K., El Saddik, A.: Gain-based Selection of Ambient Media Services in Pervasive Environments. Springer Journal Mobile Networks and Applications (2008)
20. Karmouch, E., Nayak, A.: A Distributed Constraint Satisfaction Problem for Virtual Device Composition in Mobile Ad Hoc Networks. In: Global Telecommunications Conference, GLOBECOM (2009), Current Version (2010)
21. Jiang, S., Xue, Y., Schmidt, D.: Minimum Disruption Service Composition and Recovery in Mobile Ad hoc Networks. The International Journal of Computer and Telecommunication Networking 53(10) (2009)
22. Schuster, M., Domene, A., Vaidya, R., Arbanowski, S., Kim, S.M., Lee, J.W., Lim, H.: Virtual device Composition. In: 8th International Symposium on Autonomous Decentralized Systems (2007)
23. Senthivel, K., Kalasapur, S., Kumar, M.: PerSON: A Framework for Service Overlay Network in Pervasive Environments. In: Kuo, T.-W., Sha, E., Guo, M., Yang, L.T., Shao, Z. (eds.) EUC 2007. LNCS, vol. 4808, pp. 671–682. Springer, Heidelberg (2007)
24. Park, H., Park, J.-H., Kim, N.: A Framework for Interoperability of Heterogeneous Devices in Ubiquitous Home. In: 2nd Conference on Advances in Future Internet (2010)
25. Ibrahim, N., Le Mouël, F.: A Survey on Service Composition Middleware in Pervasive Environments. International Journal of Computer Science Issues, IJCSI (2009)
26. Lee, J.W., Kim, S.M., Lim, H., Schuster, M., Domene, A.: A Software Architecture for Virtual Device Composition and Its Applications. In: Ichikawa, H., Cho, W.-D., Satoh, I., Youn, H.Y. (eds.) UCS 2007. LNCS, vol. 4836, pp. 150–157. Springer, Heidelberg (2007)
27. Grigoras, D., Riordan, M.: Service Driven Mobile Ad Hoc Networks Formation and Management. In: 4th Symposium on Parallel and Distributed Computing (2005)
28. Gu, X., Nahrstedt, K.: Distributed Multimedia Service Composition with Statistical QoS Assurances. IEEE Transactions on Multimedia 8(1), 141–151 (2006)
29. Jafarpour, H., Hore, B., Mehrotra, S., Venkatasubramanian, N.: CCD: Efficient Customized Content Dissemination in Distributed Publish/Subscribe. In: Bacon, J.M., Cooper, B.F. (eds.) Middleware 2009. LNCS, vol. 5896, pp. 62–82. Springer, Heidelberg (2009)
30. Qian, Z., Guo, M., Zhang, S., Lu, S.: Service-Oriented Multimedia Delivery In Pervasive Space. In: IEEE Wireless Communications & Networking Conference (2009)
31. Xu, C., Muntean, G.-M., Fallon, E., Hanley, A.: Distributed Storage-Assisted Data-driven Overlay Network for P2P VoD Services. IEEE Transactions on Broadcasting (2008)
32. Nahrstedt, K., Balke, W.-T.: Towards Building large Scale Multimedia Systems and Applications: Challenges and Status. In: ACM International Workshop on Multimedia Service Composition (2005)
33. The Spontaneous Virtual Network (SpoVNet), `http://www.spovnet.de`
34. Kim, J.W., Han, S.W., Yi, D.-H., Kim, N., Kuo, C.-C.J.: Media-Oriented Service Composition with Service Overlay Networks: Challenges, Approaches and Future Trends. Journal of Communications (2010)
35. Buford, J.F., Yu, H., Lua, E.K.: P2P Networking and Applications. Morgan Kaufmann (2009)