Proposition de recherche doctorale

Création de services transmédia pour la maison

Mots clés :

- Directeur de thèse : Jean-Claude DUFOURD
- Co-encadrant(s) :
- Unité de recherche : Laboratoire Traitement et Communication de l'Information
- Ecole doctorale : Ecole Doctorale Informatique, Télécommunications, Électronique de Paris
- Domaine scientifique principal : Divers

Résumé du projet de recherche (Langue 1)

Un service transmédia est un ensemble d'applications multimédia en général basées sur des technologies web et fonctionnant sur plusieurs appareils comme TV, tablette, téléphone mobile, ordinateur, box internet… La création de services « réels » dans un tel environnement est un problème très difficile. Le but de cette thèse est de supprimer une grande partie de la complexité induite par cet environnement de service pour les auteurs, de manière à permettre un élargissement de la communauté des auteurs, pour l'instant limitée à des programmeurs avec de larges compétences. Les problèmes identifiés dans la création de services sont : - la distribution hybride, c’est à dire qui utilise des médias distribués sur des canaux différents et/ou rendus sur des appareils différents, incluant des problèmes de synchronisation de la distribution des médias. Cela prendra la forme d’un outil de gestion de la distribution qui s’insère entre la création du service et son déploiement. - les plateformes hybrides avec synchronisation, c’est à dire des plateformes incluant plusieurs types différents d’appareils communicants. Sur la base d’un graphe de dépendance étendu, cross-format, un outil de partitionnement d’une application (widget) en plusieurs composants de service sera conçu. Le partitionnement sera automatique pour les situations simples et assisté dans les cas plus complexes. Un outil de co-création générant automatiquement la logique de communications entre plusieurs composants d’un service à partir de l’interface de communication sera créé. Ces deux outils aideront la transformation d’applications HbbTV vers des services Coltram. - les technologies « hybrides », c’est à dire des services créés sur base d’un mélange (fluide) de formats déclaratifs et de scripts : la tendance des « webapps » est de créer des descriptions HTML quasi-vides avec des scripts complexes et/ou du AJAX. Les deux font souvent appel à des bibliothèques populaires comme jQuery. La création de telles applications nécessite des outils complètement différents, les outils traditionnels de développement web étant inappropriés. Les outils disponibles sont des interfaces de programmation. Plusieurs stratégies seront explorées pour aider la création d’applications contenant beaucoup de script, dont la création de plugins et d’extensions jQuery, et des mécanismes de templates qui sont compatibles avec script et techniques déclaratives. L’interface DOM crée une équivalence entre déclaratif et script. Tout fragment XML peut être réécrit comme une suite d’appels DOM, et toute suite d’appels créant des éléments en DOM sur la base de constantes peut être réécrit en un fragment XML. Un outil de réécriture bi-directionel sera créé. Cette équivalence peut être utilisée de nombreuses façons : quand un auteur veut ajouter des paramètres dans un fragment XML, une traduction vers JS est nécessaire ; pour simplifier du code, il est bon d’en retranscrire les parties constantes vers XML. La suite en anglais. Transmedia services are sets of multimedia applications based on web technologies and running on multiple devices such as TV, tablet, smartphone, computer, internet gateway... Authoring of real life services in such an environment is a very hard problem. The goal of this PhD is to remove the extra complexity introduced by this environment for the service authors, so as to allow authoring by a wide audience, i.e. not just highly skilled programmers. Identified problems are authoring of service for: - Hybrid delivery, i.e. which use media delivered over different channels and rendered on different devices, including synchronisation of media delivery. This will take the form of a delivery management tool, run after service creation and before service deployment. - Hybrid platforms with synchronization, i.e. platforms with multiple different, communicating devices. Based on an extended, cross-format dependency graph, a tool for partitioning an existing gadget into multiple service components will be created; the partitioning will be automatic in simple situations and assisted in more complex ones. A co-design tool for multiple communicating gadgets will automatically generate the communication logic from the interface specification. Both tools will allow the transformation of HbbTV applications into Coltram service components. - Hybrid technologies, i.e. services implemented in a (fluid) mix of declarative and script formats. The current trend in Web-App authoring is to use very sparse HTML descriptions with complex scripts and/or AJAX. Both complex scripts and AJAX are often designed based on popular libraries such as jQuery. Authoring such applications requires completely different tools, and traditional Web authoring tools are inappropriate. Available tools are programming IDEs. Multiple strategies will be investigated to help with script application authoring, including creating jQuery extensions/plugins, script-declarative equivalence, and template mechanisms that are compatible with both script and declarative approaches. The DOM interface creates an equivalence between the two. Any XML fragment can be rewritten as a set of DOM calls, and any set of DOM calls with constant parameters can be rewritten as an XML fragment. A bi-directional rewriting tool will be created. This equivalence has many possible uses: when an author wants to “parameterize” an XML fragment, a translation to JS is required; to simplify JS code, it is good to translate any constant DOM call back to XML.

Résumé du projet de recherche (Langue 2)

Un service transmédia est un ensemble d’applications multimédia en général basées sur des technologies web et fonctionnant sur plusieurs appareils comme TV, tablette, téléphone mobile, ordinateur, box internet… La création de services « réels » dans un tel environnement est un problème très difficile. Le but de cette thèse est de supprimer une grande partie de la complexité induite par cet environnement de service pour les auteurs, de manière à permettre un élargissement de la communauté des auteurs, pour l’instant limitée à des programmeurs avec de larges compétences. Les problèmes identifiés dans la création de services sont : - la distribution hybride, c’est à dire qui utilise des médias distribués sur des canaux différents et/ou rendus sur des appareils différents, incluant des problèmes de synchronisation de la distribution des médias. Cela prendra la forme d’un outil de gestion de la distribution qui s’insère entre la création du service et son déploiement. - les plateformes hybrides avec synchronisation, c’est à dire des plateformes incluant plusieurs types différents d’appareils communicants. Sur la base d’un graphe de dépendance étendu, cross-format, un outil de partitionnement d’une application (widget) en plusieurs composants de service sera conçu. Le partitionnement sera automatique pour les situations simples et assisté dans les cas plus complexes. Un outil de co-création générant automatiquement la logique de communications entre plusieurs composants d’un service à partir de l’interface de communication sera créé. Ces deux outils aideront la transformation d’applications HbbTV vers des services Coltram. - les technologies « hybrides », c’est à dire des services créés sur base d’un mélange (fluide) de formats déclaratifs et de scripts : la tendance des « webapps » est de créer des descriptions HTML quasi-vides avec des scripts complexes et/ou du AJAX. Les deux font souvent appel à des bibliothèques populaires comme jQuery. La création de telles applications nécessite des outils complètement différents, les outils traditionnels de développement web étant inappropriés. Les outils disponibles sont des interfaces de programmation. Plusieurs stratégies seront explorées pour aider la création d’applications contenant beaucoup de script, dont la création de plugins et d’extensions jQuery, et des mécanismes de templates qui sont compatibles avec script et techniques déclaratives. L’interface DOM crée une équivalence entre déclaratif et script. Tout fragment XML peut être réécrit comme une suite d’appels DOM, et toute suite d’appels créant des éléments en DOM sur la base de constantes peut être réécrit en un fragment XML. Un outil de réécriture bi-directionel sera créé. Cette équivalence peut être utilisée de nombreuses façons : quand un auteur veut ajouter des paramètres dans un fragment XML, une traduction vers JS est nécessaire ; pour simplifier du code, il est bon d’en retranscrire les parties constantes vers XML. La suite en anglais. Transmedia services are sets of multimedia applications based on web technologies and running on multiple devices such as TV, tablet, smartphone, computer, internet gateway... Authoring of real life services in such an environment is a very hard problem. The goal of this PhD is to remove the extra complexity introduced by this environment for the service authors, so as to allow authoring by a wide audience, i.e. not just highly skilled programmers. Identified problems are authoring of service for: - Hybrid delivery, i.e. which use media delivered over different channels and rendered on different devices, including synchronisation of media delivery. This will take the form of a delivery management tool, run after service creation and before service deployment. - Hybrid platforms with synchronization, i.e. platforms with multiple different, communicating devices. Based on an extended, cross-format dependency graph, a tool for partitioning an existing gadget into multiple service components will be created; the partitioning will be automatic in simple situations and assisted in more complex ones. A co-design tool for multiple communicating gadgets will automatically generate the communication logic from the interface specification. Both tools will allow the transformation of HbbTV applications into Coltram service components. - Hybrid technologies, i.e. services implemented in a (fluid) mix of declarative and script formats. The current trend in Web-App authoring is to use very sparse HTML descriptions with complex scripts and/or AJAX. Both complex scripts and AJAX are often designed based on popular libraries such as jQuery. Authoring such applications requires completely different tools, and traditional Web authoring tools are inappropriate. Available tools are programming IDEs. Multiple strategies will be investigated to help with script application authoring, including creating jQuery extensions/plugins, script-declarative equivalence, and template mechanisms that are compatible with both script and declarative approaches. The DOM interface creates an equivalence between the two. Any XML fragment can be rewritten as a set of DOM calls, and any set of DOM calls with constant parameters can be rewritten as an XML fragment. A bi-directional rewriting tool will be created. This equivalence has many possible uses: when an author wants to “parameterize” an XML fragment, a translation to JS is required; to simplify JS code, it is good to translate any constant DOM call back to XML.
The Coltram project is about offering services composed of multiple media on a dynamic set of devices within the home, i.e. dynamically mapping user-oriented services onto the multi-polar home network, to achieve the best possible quality of experience for the users at any time. One key component of the home environment is the connected TV running interactive TV services. Coltram will design, implement and promote a Coltram service model providing such distributed services on the heterogeneous home network taking into account security, changing availability of devices and user request for service reconfiguration.

In order to support this service model, a model for dynamic deployment of services onto the changing home environment is required. The generic term service comprises here applications that interact with users (in the foreground), applications that do not interact with users but provide services (in the background), as well as rather static service provided by devices features (e.g. "switch channel"). The model will support initial deployment of services, as well as redeployment when resources change or upon user request. There will be a close cooperation with the EU (FP7) funded project "webinos", www.webinos.org, which targets on a cross-platform application infrastructure focussing on the four domains: mobile, automotive, desktop, and home media. As webinos focuses on the actual execution platform and the technical design of an application, Coltram will address the not covered, but equally important aspects of the authoring of the application and services – especially to reflect the individual characteristic of the envisioned devices in the home media domain. In this sense Coltram can benefit (build on) from webinos results and community delivering additional urgently needed solutions. Additionally, Coltram will be able to provide insights and recommendations to webinos, to influence it’s direction and to ensure a better joint exploitation. The Coltram model induces extra complexity for the author, on top of the complexity of widget authoring. To compensate this, one part of the project is dedicated to making the creation of Coltram services easier. The following authoring aspects will be considered: authoring services consisting of a constellation of communicating web and native components, e.g. using synchronized media coming from multiple delivery mechanisms, such as broadcast and broadband; authoring of applications that can make the application usage experience more dynamic and reduces the application development effort. - adapting a service may be executed on a choice of devices, and as such, may need to be adapted. A service, on the other hand, should be easy to reuse, previous designs, and with facets for different authoring expertise levels, including one facet for a wide audience.}

Today, any service to the user is either implemented as a (native) software application, running on a particular device and OS, or if implemented as a Web application, then as one document (view/session) at a time and coming from the Internet (online). The home environment is constituted of more and more devices with extreme varying characteristics. It is heterogeneous, whatever the point of view: devices have a screen or not, large or small, have input capabilities or not, are personal and public, have computing power or not... Such an environment is, in a sense, multi-centralized. Specific features of a device implicitly make this device a server for these specific features. Any service requiring these features must connect to that device or to a service exported by that device. Parts of the service may be executed on a choice of devices, and as such, may need to be adapted. A service, on the other hand, should be easy to use. Detailed management of its mapping onto devices should be transparent to the user. More: if the availability of devices changes, then the service should be transparently reconfigured, without loss of execution context. Coltram aims at a new model of service, which is a collaboration of multiple applications running on multiple devices, including discovered devices, each application not tied to one device and able to move to another device without losing state, and coming from Internet or broadcast or the local network. And yet, because native software applications will stay as an important component of the home architecture, the new service model should allow the seamless integration of native and Web components, and the seamless switching between native and Web components. The challenge is to allow the convergence of the mobile, the internet world, the home media (TV, set top boxes, gaming consoles, etc.) world within the home network by providing this new service model, lowering the cost of designing and using these services for all actors. [[(State of the art)]]. In the WWW, services are implemented with Flash or Web standards, with one server for all the services, one set of APIs, one model for the service and how to interact with the service, one way of authoring the service, one way of publishing and deploying the service. Emerging standards such as HbbTV in the TV domain are pointing toward a similar model (without Flash or dedicated apps). The home network however is quite different: there are many different devices, many of which are not always available; communications are not naturally centralized, but device-to-device; but Web standards are a common base for the design of services if peer-to-peer discovery and communication interfaces can be added. Services on Internet or on mobiles are designed to run on one single, relatively powerful device, which has computing power and a screen/speakers and interaction capabilities, as well as possible extra features such as a camera. In the home network, capabilities required by a service are rarely concentrated on a single device: usually there are multiple choices for each requirement, and the best available choice may change from moment to moment. Concepts from various origins need to be merged: ubiquitous/pervasive computing, where an application can run on any device; distributed computing, where a service may be implemented as several communicating applications running on different devices; and Web applications, where the service is implemented as a document presented in a Web browser (with script). Pervasive computing addressed the various problems of mapping applications onto heterogeneous sets of devices. Service adaptation was studied in multiple contexts. However, there are no solutions yet addressing the Coltram combination of features: services implemented as constellations of communicating Web applications; Web applications can be split so that a device-dependent part can stay on that device; devices come and go, and applications may need to be moved during the life of the service while keeping the service state. As its basic framework Coltram will build on the webinos framework, which is a federated Web runtime that offers a common set of APIs to allow apps easy access to cross-user cross-service cross-device functionality in an open yet secure manner. Other words, webinos provides the secure Web application framework on top of which the Coltram service model can be built. [[(Scientific and technical objectives of the project)]]. The scientific and technical objectives of the project are: - to design, implement and promote a Coltram service model providing distributed services on an heterogeneous home network including the TV, taking into account security, changing availability of devices and user request for service reconfiguration; - to design, implement and distribute service production tools that will significantly reduce the cost and complexity of services running on the home network; - to provide the above implementations as free, open source software; - to enhance the dissemination of the models by their inclusion in open standards. The technological obstacles are: - there is no efficient standard for discovery and service communications interface with Web standards, nor for moving sessions (documents) around while keeping their state; webinos will provide a basic service type based discovery API for Web application suitable service filter as well as a more open discovery API which allows to define goals instead of concrete APIs will not be covered by webinos; - there is no model of the home environment as a multipolar set of devices with some with specific features, i.e. a set of devices without a clear center, with a focus on different devices depending on the point of view (computing, display, interaction, privacy, etc); - service partitioning to run as multiple documents on multiple devices is a hard problem; webinos defines the concept of child applications which are small applications nested in a main application package to design applications for distributed usage. The nested applications can be deployed on demand to other devices more suitable for a given task. However, in webinos this is purely application driven. I.e. the application developer has to choose when which parts should be executed remotely. Coltram will develop more advanced techniques to allow the system to automatically distribute application logic that can make the application usage experience more dynamic and reduces the application development effort. - adapting a service, with a particular home configuration is a new and hard problem; - providing security without impacting the service usability is almost an oxymoron; - there are no tools for authoring multi-device and multi-delivery services; - the authoring complexity (and cost) is very high, and this new service model makes that even worse; - the device capabilities of the different categories are quite disjoint:
whereas mobile phone have become very powerful, but have shortcomings in battery, TV sets or set top boxes have limited processing and I/O capabilities (or primarily optimized for video play-back only); # the usage paradigms of the different device categories are quite different: whereas the mobile phone is regarded as a very personal trusted and interactive devices, a TV set is still a shared (even privately) lean-back device; # especially in home media domain the plug-'n-play interworking without any (or little as possible) administration need is an absolute requirement and key to success. As this is sufficiently implemented for the exchange of multimedia content (standardized via UPnP or even proprietary within a vendor’s ecosystem), this is one of toughest challenge for cross-device applications. {{Scientific approach}} Coltram, in coordination with webinos, within a new W3C working group, will design and promote a unified discovery and service protocol interface, allowing applications (Web or native) to interface transparently with UPnP, Bonjour, SLP or other/future discovery and service protocols. Coltram will design a new model for the home environment as a multipolar set of devices with specific features and provide tools to partition and adapt services to run as multiple communicating entities on the multipolar home network, taking into account device specificities and the dynamic side of the home network. The Coltram service model will be of a set of communicating applications. Each application can be dynamically moved to the “best possible device” at any point in time, if its required features do not tie it to one device – and even then, only those parts of an application that specifically require the feature might be tied to the device, with other parts being dynamically moveable. The model for communicating applications will be that of widgets (read packaged Web applications: http://www.w3.org/TR/widgets/) extended with discovery and service capabilities, reaching forward in the direction of the work of the Home Network Task Force of the W3C Web and TV interest group. There are two approaches to the deployment of Coltram services: centralized and decentralized. The centralized version will be developed on top of the webinos “personal zone hub” platform. An alternative, more innovative, decentralized approach will be entirely designed in Coltram. Coltram will create an authoring tool chain and platform that supports distributed application development by design with the aim of making explicit application development for specific platforms or devices unnecessary. This includes new application packaging and deployment concepts to ensure seamless deployment of distributed applications to all devices needed for application execution. Thus, the user is not required to install application components on each of the incorporated devices, e.g., by going to application stores on each device.

Informations complémentaires (Langue 1)

This PhD will be funded by a COLTRAM, a 3 year collaboration program between Fraunhofer FOKUS in Berlin and Telecom ParisTech in Paris. The Multimedia/GPAC team has a long experience of the transmedia environment, of open source software (GPAC player) and of standardisation. We have participated in many national and industrial projects on connected subjects, in particular openHbb to implement HbbTV, a project with Samsung on communicating widgets, projects on digital radio… The European project webinos, http://webinos.org, of which the team is an « affiliate member », defines an environment with solutions for many of the stated problems. Platforms addressed by webinos are PC (Mac, Linux, Windows) Android and the STB and automotive platforms.