

# Flexible Services LUCRE

## Releases of LUCRE tools and system

Authors:

Pia Ojanen, Department of Media Technology, TKK

XIDE: <http://flexi.tml.hut.fi/xide/#ap>

Service Composer: <http://flexi.tml.hut.fi/ospace>

# Table of Contents

<b>1. Vocabulary.....</b>	<b>4</b>
<b>2. LUCRE Uspace Prototype .....</b>	<b>4</b>
1.1. System Description.....	5
1.2. Service Structure.....	5
1.3. System Requirements.....	6
1.1.1. Desktop.....	6
1.1.2. Mobile devices.....	6
1.4. Service Composer architecture.....	6
1.5. Server Software.....	7
<b>3. Implementation of Uspace.....</b>	<b>8</b>
1.1.3. Including widget to a shared space.....	8
1.6. Space views.....	8
1.7. Mashup between widgets.....	8
<b>4. Widgets .....</b>	<b>9</b>
1.8. Wookie Widget Engine.....	9
1.9. Widgets.....	9
1.1.4. Gift Planner widget.....	9
1.1.5. Other widgets.....	9

# List of Figures

FIGURE 1: WORKFLOW .....	6
FIGURE 2: SERVICE ARCHITECTURE .....	7

# 1. Vocabulary

- USpace
  - o Contains one or more Spaces
  - o User specific
- Space
  - o Visible to a specified user group or to everyone
  - o One per service
  - o Consists of Subspaces, Widgets and Documents
- Subspace
  - o Belongs to a certain Space
  - o Consists of Widgets and Documents
- Service Composer
  - o Creates and modifies Spaces and Subspaces
    - Users can add widgets
    - Users can edit widgets
    - Users can combine widgets
    - Users can edit documents
- Widget
  - o Portable piece of code that can be executed on any web page without requiring additional compilation
  - o Users can create, modify and combine widgets
- Document
  - o HTML –like document
  - o Contains widgets among the others
  - o Possible to create document widgets
- Widget Engine
  - o A software system running widgets
- XIDE (XFormsDB IDE)
  - o A Graphical editor for creating small web applications and widgets
  - o Users can create, modify and combine Widgets
- Canvas
  - o Work area in Service Composer

## 2. LUCRE Uspace Prototype

We decided to call LUCRE shared group workspace Uspace for the time being. Name is not final, but it will be used in this document for clarity.

In this preliminary prototype the entire workflow is not available. First version presenting the whole workflow will be released 26.6.2009. First user trials could be started in the end of August 2009. Very simple version of the prototype is available at: <http://flexi.tml.hut.fi/ospace>

In the future there will be a major release in every half a year.

## **1.1. System Description**

Uospace is a collaborative workspace for multiple users, which takes advantage of the location information. It should be highly expandable and it should offer possibilities to users with varying computing skills.

End-users with no programming skills should be able to compose entire services consisting of documents and widgets using LUCRE tools. Social dimension should be present in all levels, which means that spaces and widgets can be shared with friends, groups or communities.

More advanced users should be able to create and modify widgets. Web based IDE called XIDE can be used for creating widgets, but traditional DHTML widgets are supported as well. Thus widget developers can choose between coding by hand and using a visual IDE for service creation.

## **1.2. Service Structure**

Entire workflow is a combination of systems that are created in EDEN and LUCRE. Figure 1 presents the process from beginning to composition and mash-up between widgets.

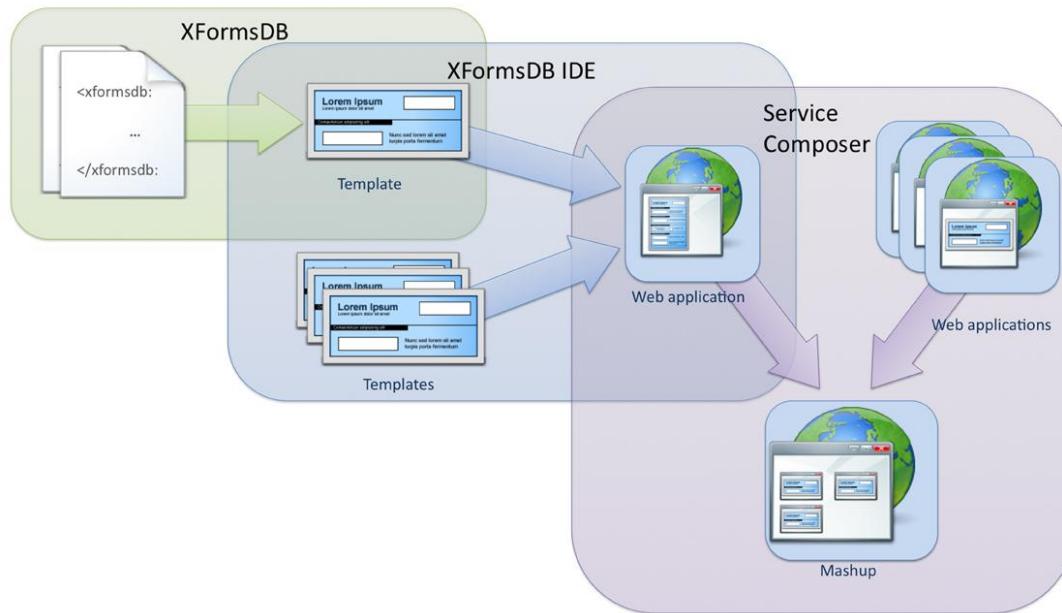


Figure 1: Workflow

### 1.3. System Requirements

#### 1.1.1. Desktop

Service composer is usable on all last generation Internet browsers. The service has been tested with Mozilla Firefox, Microsoft Internet Explorer, Apple Safari, Google Chrome and Opera. Some browser dependencies might be present in early prototypes.

JavaScript support should be turned on, as the implementation relies much on the newest JavaScript techniques.

#### 1.1.2. Mobile devices

Support for mobile services will be implemented as a part of this project later for the first user trials.

### 1.4. Service Composer architecture

The System architecture is based on the client/server model.

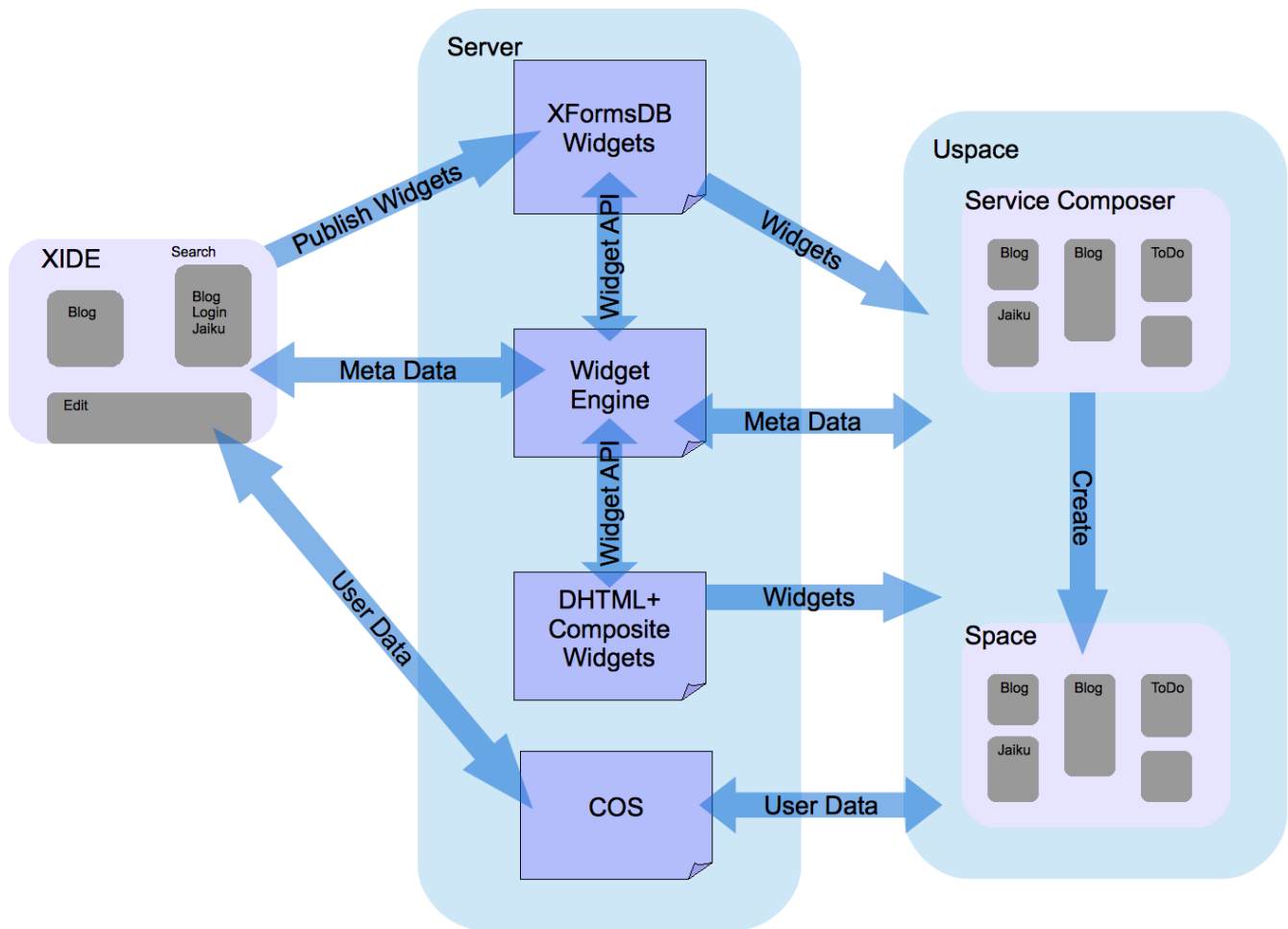


Figure 2: Service architecture

## 1.5. Server Software

Server contains most of application logic in the system. It contains following software components:

- Servlet Container
  - o Hosts web applications
  - o Technology: Apache Tomcat (<http://tomcat.apache.org/>)
- Uspace Database
  - o MySQL
- Client UI
- Wookie Widget Engine
- Wookie Database
  - o MySQL
- XFormsDB Widgets
- DHTML Widgets

## 3. Implementation of Uspace

In this stage of implementation the elements described later are not yet functioning fully. Work with user interface and widgets has started and progressing steadily.

Uspace has separate editing state for each space, called Service Composer. In normal viewing state canvas consist of widgets and written content. When user presses the edit button the service composer will be opened and user can drag and drop all textual fields, add and combine widgets, edit widget properties and so on.

Business logic for Uspace UI will be implemented with Spring MVC framework and all modern user interface elements will be implemented using jQuery and Ajax.

### 1.1.3. Including widget to a shared space

Widgets will be embedded to the code using object tags. In this way widgets can be run in any server. The only requirement is that widget can be referenced by an URI and knowledge about user and space information.

```
<object data="http://server/component.xformsdb" width="x" height="y" type="text/html"></object>
```

When user wants to add or edit widgets the search window will be opened to the right side of the canvas. In final implementation user can drag-and-drop widgets from the panel to the canvas and after that the element is editable in the work area. In the future user can also combine widgets in this state.

### 1.6. Space views

Views in different formats could be defined internally in the Service. Different views will be grid view, document view, absolute view and mobile view. User can switch between views from the user interface.

User should be able to preview service in mobile view from desktop view. Mobile view could be edited in desktop environment. User cannot edit the service itself from mobile view. Content of the widgets and documents can be modified from all views.

Multiple Space views will be implemented later. Implementation has been started from the grid view.

### 1.7. Mashup between widgets

It is possible to merge services in two ways. First option is one time information exchange from one widget to the other during the execution. Second one is merging two existing widgets to a new permanent widget.

Users could combine widgets by dropping one widget on top of another.

Mashup between will be implemented later.

## 4. Widgets

Widgets for Uspace will be created as a part of LUCRE. Widgets are managed with open source widget Engine.

### 1.8. Wookie Widget Engine

<http://getwookie.org/Welcome.html>

Wookie Engine is an open source server application that allows users to upload and deploy widgets. It allows widgets make calls to other sites, APIs and feeds via HTTP a proxy. Wookie is based on the W3C Widgets specification.

Wookie server is a Java Servlet application that requires a MySQL database and a Java Servlet container such as Apache Tomcat.

### 1.9. Widgets

Wedding planner was selected as the first actual use case for LUCRE project. Widgets will be implemented with XFormsDB language, using XIDE development tool when it is ready. First widgets have been coded by hand.

#### 1.1.4. Gift Planner widget

Gift planner widget has been implemented as a part of LUCRE wedding use case.

Single sign on system is not yet implemented, so user has to log in using following user names and passwords:

User name: user - Password: pass

Admin name: admin - Password: pass

#### 1.1.5. Other widgets

Other widgets that are included in the prototype have been coded as a part of EDEN project earlier.

CNN news  
Simple blog widget