SOCIAL WEAVER

A PROTOTYPE FOR WEAVING WEB 2.0 FEATURES INTO WEB APPLICATIONS

Supervisor:
Dr. Michael **Felderer**

In collaboration:
Dr. Dirk **Draheim**

Masterseminar
QE
29.04.2013

By:
Viktor **Pekar**

AGENDA

- Theoretical Background
  - The Problem is the Motivation
  - The Solution is Social Weaving
- Explanation of Social Weaving
  - Possible Use Case (Recap)
- Presentation of the Social Weaver Prototype
  - Architecture
  - Implementation
  - Live Demo
- Problems with the Web Culture
  - Perfect but hardly reachable Solution
  - Alternatives

THEORETICAL BACKGROUND

- Theoretical Background
  - The Problem is the Motivation
  - The Solution is Social Weaving
- Explanation of Social Weaving
  - Possible Use Case (Recap)
- Presentation of the Social Weaver Prototype
  - Architecture
  - Implementation
  - Live Demo
- Problems with the Web Culture
  - Perfect but hardly reachable Solution
  - Alternatives

Social Weaver Prototype

THEORETICAL BACKGROUND
THE PROBLEM

- Using the internet and web applications to **achieve (and simplify) tasks** (in work, studies, everyday life, ...)
- What if the web application becomes a problem itself?
- **Communication is hard** about specific elements in the system or workflows – especially when working remotely

Social Weaver Prototype

THEORETICAL BACKGROUND
THE SOLUTION

- **Inject** (or how we call it: weave) **social web elements** into the web application that allows users to discuss about and interact with it
  - without modifying the underlying code though...
- Direct relation between communication thread and relevant element or workflow

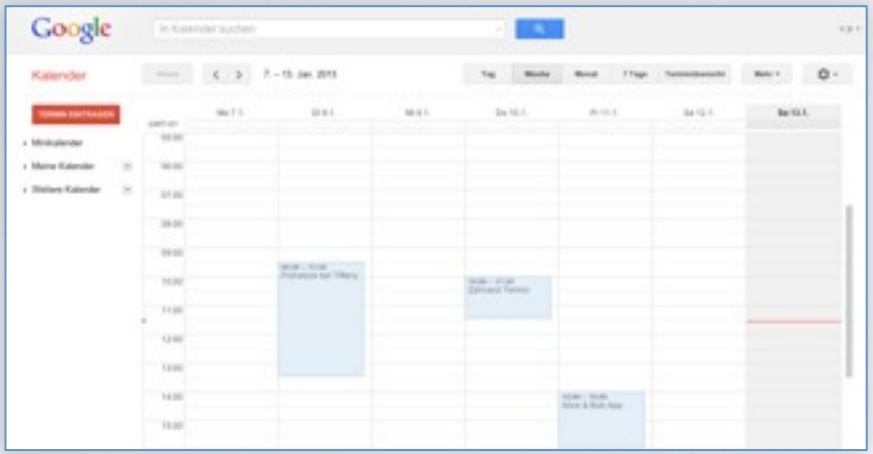
Social Weaver Prototype

SOCIAL WEAVING

- Theoretical Background
  - The Problem is the Motivation
  - The Solution is Social Weaving
- Explanation of Social Weaving
  - Possible Use Case (Recap)
- Presentation of the Social Weaver Prototype
  - Architecture
  - Implementation
  - Live Demo
- Problems with the Web Culture
  - Perfect but hardly reachable Solution
  - Alternatives

Social Weaver Prototype

USE CASE FOR SOCIAL WEAVING



- Two Google Calendar users, Alice and Bob, use a shared calendar
- Alice wants to ask Bob something about their date that is happening Friday night.
- instead of calling or texting Bob, Alice uses a social weaving feature and adds a comment box to the appointment
- Next time Bob opens his Google Calendar – he sees the comment box attached to the appointment and can answer the question.

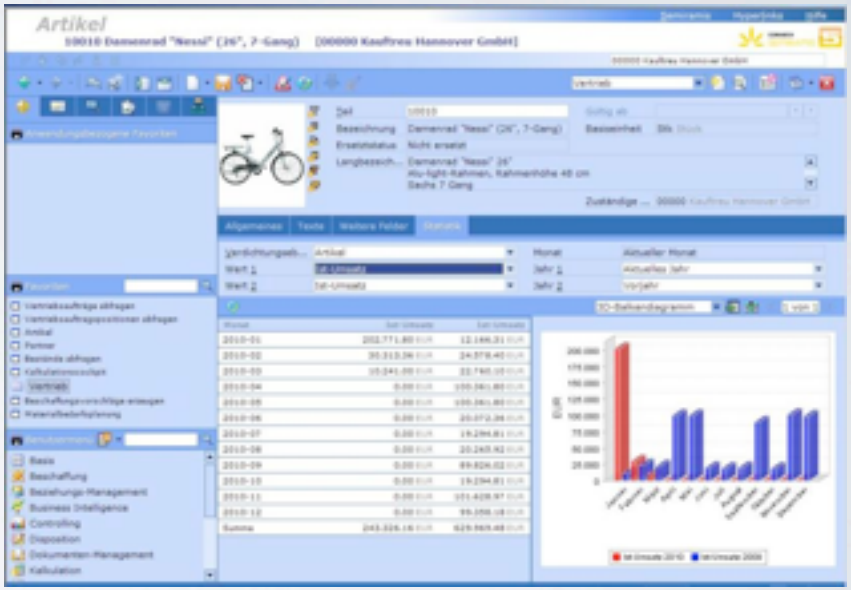
Social Weaver Prototype

ALICE AND BOB – USE CASE IN OUR CONTEXT

- Social Weaving is way more than just about Google calendar or comment boxes

- Ultimate goal is a generic system that can attach many **social web elements** (like wiki pages, document attachments, comment boxes, chats, …) to any web application, even browser based ERP-systems

- More realistic goal (for now) is to **proof of concept** that this feature is possible to implement



Social Weaver Prototype

PROTOTYPE

- Theoretical Background
  - The Problem is the Motivation
  - The Solution is Social Weaving
- Explanation of Social Weaving
  - Possible Use Case (Recap)
- Presentation of the Social Weaver Prototype
  - Architecture
  - Implementation
  - Live Demo
- Problems with the Web Culture
  - Perfect but hardly reachable Solution
  - Alternatives

INTRODUCING
SOCIAL WEAVER

Social Weaver (Philetairus socius) is a species of bird in the Passeridae family endemic to Southern Africa

ARCHITECTURE

- Client - Server based architecture

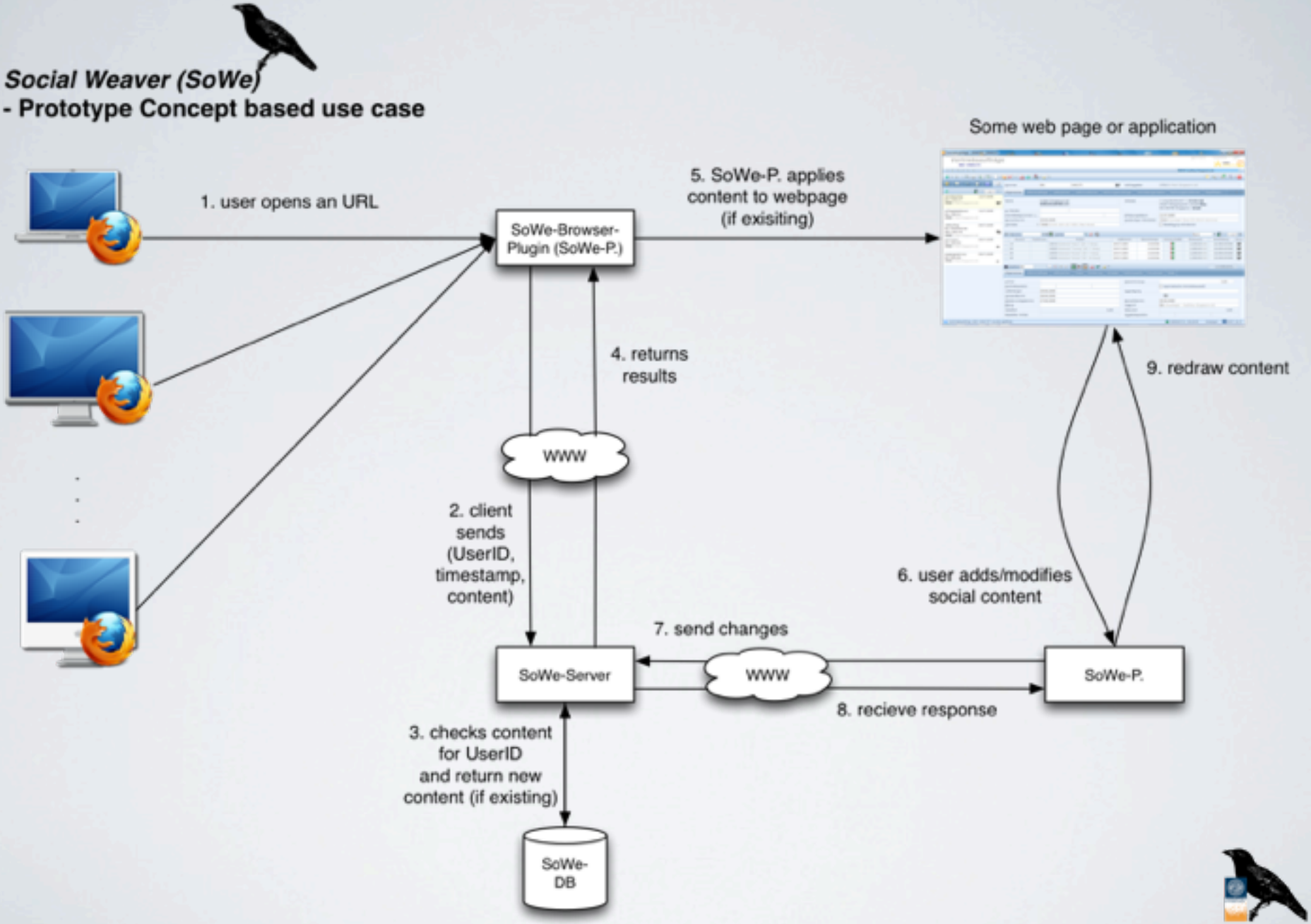
- Client - Firefox Plugin
- Server - REST Web Service with PostgreSQL

REST WEB SERVICE

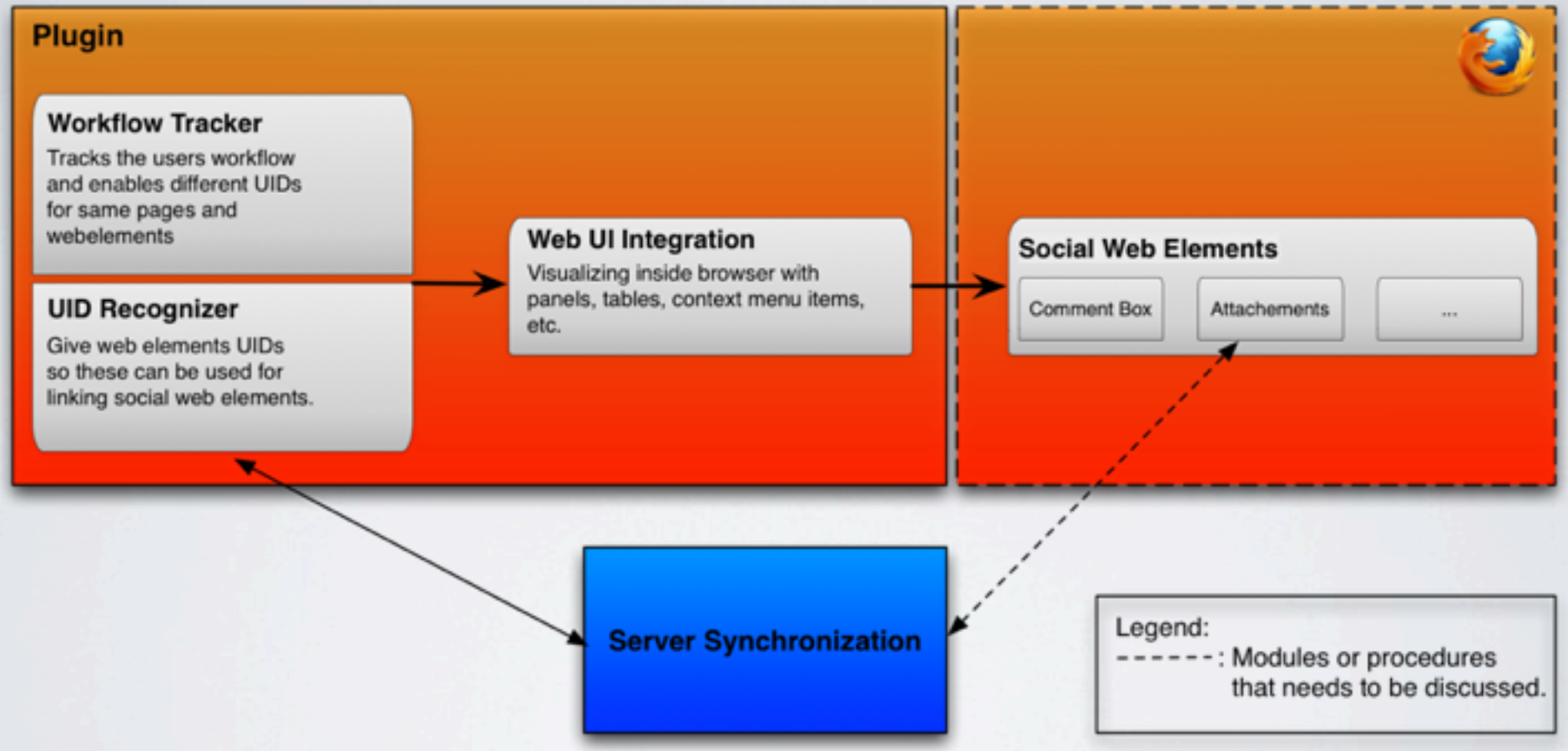
- The Web Service (WS) offers REST interfaces to receive and push Anchor information
  - **Anchor** is a data tuple that identifies an element in the web session
- Server synchronizes Anchors from different user sessions and keeps them up to date
- Server is decoupled from the client (which means that it does not distinguish whether its a Firefox plugin or something else)

CLIENT PLUGIN

- Client Plugin is implemented with the Mozilla Addon-SDK for the Firefox browser
- **Functionality**
  - Weaving social elements into websites
  - Creating and matching Anchor information to websites
  - Communicate with the REST WS


Social Weaver (SoWe)
- Prototype Concept based use case
Some web page or application
1. user opens an URL
SoWe-Browser-Plugin (SoWe-P.)
5. SoWe-P. applies content to webpage (if exisiting)
4. returns results
9. redraw content
WWW
2. client sends (UserID, timestamp, content)
6. user adds/modifies social content
7. send changes
SoWe-Server
WWW
SoWe-P.
8. recieve response
3. checks content for UserID and return new content (if existing)
SoWe-DB

Social Weaver (SoWe)
- Module Overview



LIVE DEMO

PROBLEMS WITH WEB STANDARDS

- Theoretical Background
  - The Problem is the Motivation
  - The Solution is Social Weaving
- Explanation of Social Weaving
  - Possible Use Case (Recap)
- Presentation of the Social Weaver Prototype
  - Architecture
  - Implementation
  - Live Demo
- Problems with the Web Culture
  - Perfect but hardly reachable Solution
  - Alternatives

NO OBJECT IDENTIFIERS (OID)

Simple HTML

RoboCup German Open

Dynamic Links

Jetzt individuelle & limitierte Trends nur im Onlineshop erhältlich!

Web Application Element (GCal Appointment)

<dl style="height:35px;border-color:#9FC6E7;background-color:#E4EFF8;color:#777777;" class="cbrd"><dt style="background-color:;">13:00 – 14:00 </dt><dd>(Kein Titel)</dd>   </dl>

Just a few samples for element types in the web...

WHERE OIDS ARE USED

- ERP
- JavaEE (JPA, Hibernate, ...)
- OOP

- In principle web elements have OIDs defined by its tree!
  - But hard to determine from client site...

ALTERNATIVE - TINKER FAKE OIDS...

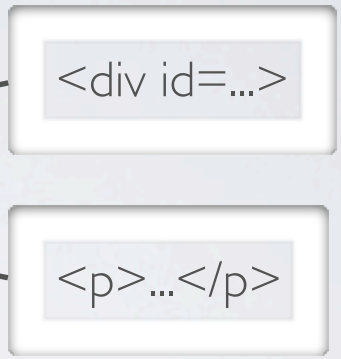
- We assume that most elements in web applications are clearly defined to the users view
- Therefore it should be possible to extract some information from the code that defines the element. (We called this Anchor previously.)
- Obviously there is **no generic solution** for all web sites and applications
  - **External scripts** are the solution to support specific environments
  - Those scripts contain information about how an anchor should be constructed

Social Weaver Prototype

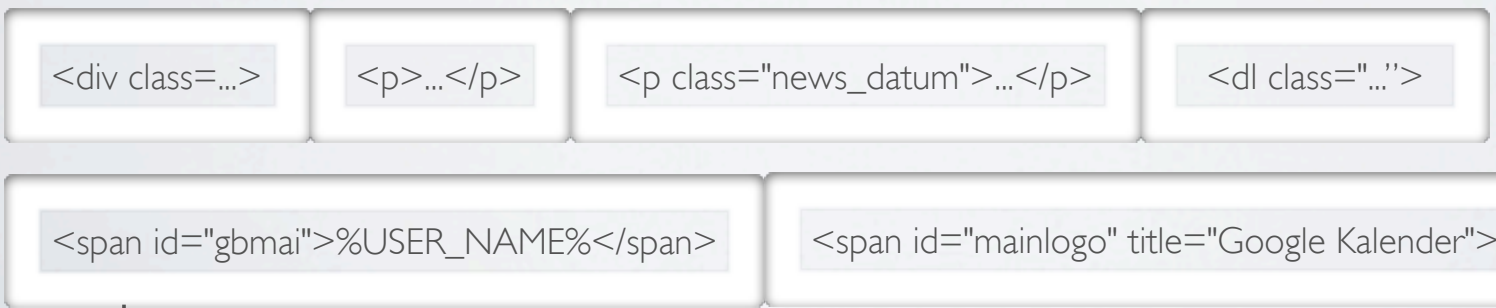
ANCHOR SCRIPT EXAMPLE

At the moment the prototype Anchor consists of three parameters:

```
this.url = anchor[0];
this.ancestorId = anchor[1];
this.anchorText = anchor[2];
```



Scripts are sets of matchable strings which are used to create Anchors. More possible strings would be:



and so on...

SCRIPT FOR GOOGLE CALENDAR

- Simple Anchor scripts will be sufficient for simple HTML sites

- Web applications will need more complicated scripts
- A script that supports google calendar will need the following parameters:

```
<span id="mainlogo" title="Google Kalender">
```

```
<span id="gbmai">%USER_NAME%</span>
```

```
<tr class="wk-daynames">
<th scope="col" title=%DAY_DATE%>
```

```
<span class="evt-lk ***">%TITLE%</span>
```

```
<dl class="cbrd">%TIME%
```

To give full support for week, day, list views etc. way more parameters would be needed...

Social Weaver Prototype

NOT SO BAD AS IT SOUNDS

- This script idea has many **disadvantages**
  - Extra effort for every web application
  - If the web application changes then the script might not work anymore
  - Hard to develop, debug and test

- But those scripts can be written by a a **automatized script generator**
  - If a user marks an element - he creates automatically a script that contains information about how to match it
  - No generic solution as well - but less work that needs to be done by hand...
  - Not the topic of this presentation today - and not even of my thesis :-)

Social Weaver Prototype

THANKS FOR YOUR ATTENTION

☺

Social Weaver Prototype