

exove

exove

Welcome!

exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvälääkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	

exove

Always select a proper
tool for the task at hand.

exove

Know what you need and
get a system that fits your
needs.

exove

When in doubt, ask help
from a professional.

exove

About us

exove

Exove delivers innovative, reliable and business-driven web services that enable our customers to conduct better business on the Internet.

exove

We base our work on our customers' strategy and needs that we develop into web services.

exove

We deliver sites, community platforms, and Web 2.0 applications.

exove

19 people, 62 customers,
1.2 MEur turnover 2008, profitable

exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvälääkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	

exove

CodeIgniter in brief

An open source web application framework

Based on model-view-controller model

- More about this on the next slide

Allows you to focus on the functionality of your system by providing all needed building blocks

www.codeigniter.com

exove

Model-View-Controller design pattern

Design patterns are elegant solution templates to certain problems

Model-View-Controller (MVC) is a design pattern for application development

- Helps the developer to separate content, control, and presentation from each other
- Makes solutions easier to understand and to modify

exove

MVC basics

Application consists of controllers

- Contain the business logic of the application
- In web applications, user request is received by a controller

Controllers use models to manipulate data

- Models create the structure of the “world”
- Hide the database from the business logic

Views display the data fetched from models based on commands from controllers

- Web application views are typically page templates

exove

MVC benefits

Makes the code easy to understand and to modify

- Controller methods are typically short
- Parse request, get data from a model, modify it (if needed), and pass it to view

Models can be reused

- Reduces complexity and footprint

When models are done right, new functionality can be implemented really fast

Copy texts and images are isolated to views

- Helps to change the content comparably easy

exove

CodeIgniter strengths

Small footprint

Good performance

- Short path to your code

Almost zero configuration

Easy to learn

- Good documentation
- Logical

Extensible and hookable

Code is split properly into directories and average file size is relative small

exove

CodeIgniter weaknesses

Application code is inside CodeIgniter code

- Framework should be inside its own directory for easier upgrading

A bit weird object-oriented model

- Never seen similar anywhere else
- Takes sometime to get used to, but it is logical

Using PHP as the template language

- Leads logic creep to templates easily
- Is butt-ugly to read

No readymade caching mechanisms

No content management system

Somewhat poor language support

- Cumbersome

exove

Exovelgniter

Exovelgniter is an extension / superset of CodeIgniter that addresses a number of our needs

All Exove web application projects are based on Exovelgniter

Major changes

- Application and framework separated
 - Typically projects symlink to the Exovelgniter code
- Better template engine, Smarty
- Multichannel support
- Multidomain support

Added features

- Automatic form handling using Former tool
- Secure login and logout mechanism
- User primitives
- Better logging
- AJAX / JSON connectivity
- WordPress bridge
- Utility classes
 - Finnish bank payment buttons
 - Image manipulation
 - Video manipulation
 - Crypting

Some minor fixes, too

exove

Smarty

Smarty is a PHP based template language

Used to convert framework internal data objects to HTML

Has simple yet powerful syntax

- Can be used by HTML codes to implement the templates
- Some programming experience needed, but not much

www.smarty.net

exove

Creating applications with CodeIgniter

exove

Before coding...

Make sure that you know what you are going to do

- Have a decent understanding of the software in general and the part under implementation in specific

Design the database

- If you can't, consider doing more thinking

Have a good concept design for the application

- Visuals are helpful, too

Application can be implemented in two ways

exove

Bottom up

Design and implement models first

Implement business logic in controllers

Implement views and JS

Check interaction and compare to the concept

Top down

Implement bare controllers first

Implement views and JS

Check interaction and compare to the concept

Implement controllers and models together by filling the blanks in the controllers

exove

Comparison of styles

Bottom up methodology suits for data intensive systems

- Hard parts are done first, and views are just the icing

Top down is better for user experience focused systems

- Allows the design team to iterate with the system before models are locked down

exove

Extending CodeIgniter

exove

Extensions

CodeIgniter is designed to be extensible with several mechanisms

- System provides eight hooks for plugging in code in the execution flow
- Core system classes can be swapped or extended
- System libraries can be extended or implemented

There are a lot of third party extensions available

ExoveIgniter uses all mechanisms

- Has also a couple of modifications inside CodeIgniter

exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvääkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	



hyvääkäri.fi
Kvalion Oy

Exove
2.12.2009

Pia Koivu

Kvalion Oy

Business Concept

- Beneficial and high-quality information about health care services for Finns.

Expertise

13 partners

- medical & health care
- private health care services
- business administration & development, economics
- communication

www.hyvalääkäri.fi

Plan A

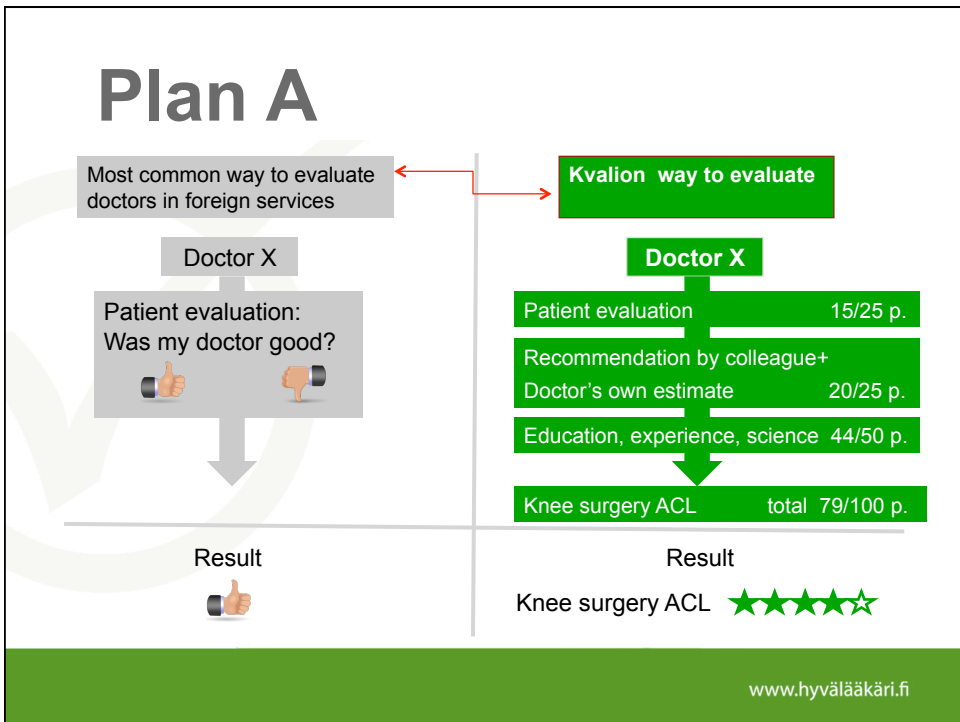
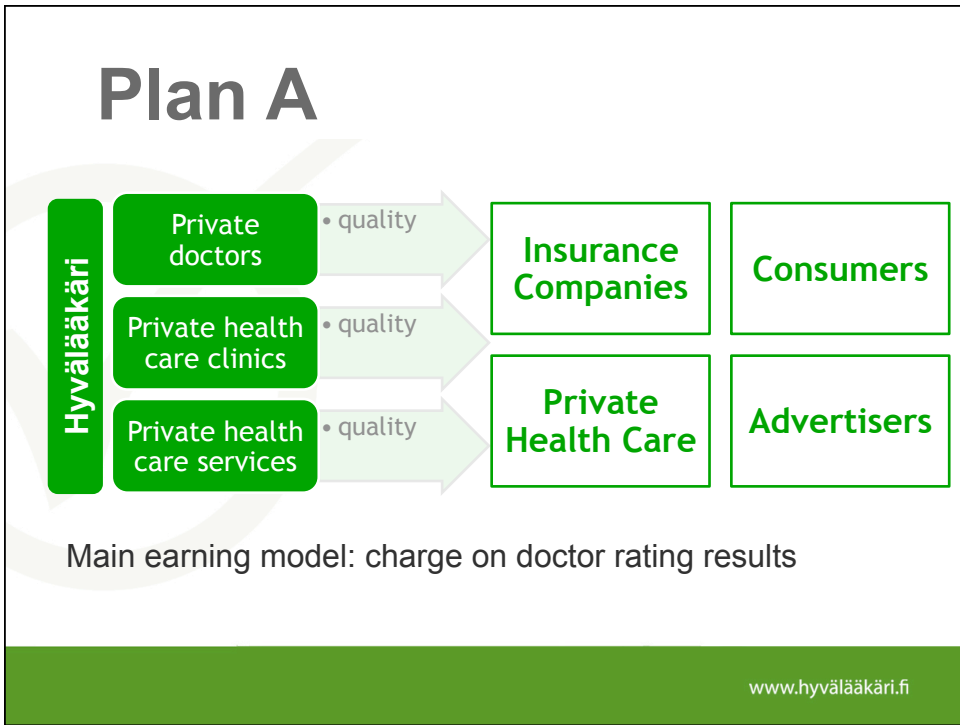
Goal

- Help consumers to find best possible doctor & care
- Improve quality of health care

Means

- Create a way to measure doctor's quality of service with an objective and fair evaluation method
- Idea used abroad. Kvalion aimed to create a more high quality and comprehensive evaluation method ever used.

www.hyvalääkäri.fi



Challenges

Recognized

- New, emotion raising, old and dusty structures/hierarchy shaking service
- Earning model
- Authorities, unions, opinion leaders, press, politicians...

Tackled

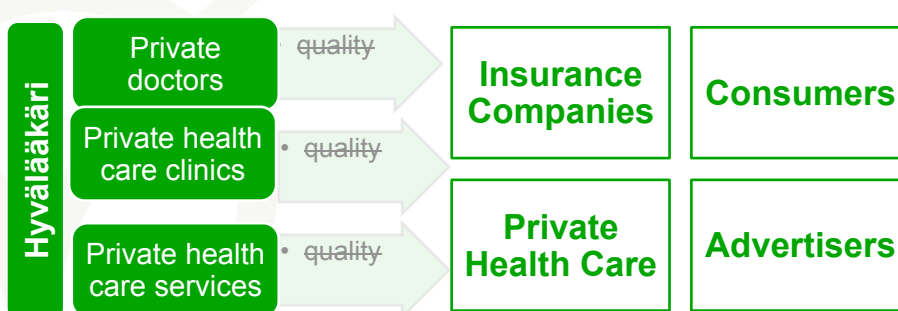
- Personally, with support by experts, by communication

Prepared for

- A > B

www.hyvalääkäri.fi

Plan B



www.hyvalääkäri.fi

Kvalion Oy

Gathering and refining information of
doctor's know how and health care
services for various industries

www.hyvalääkäri.fi

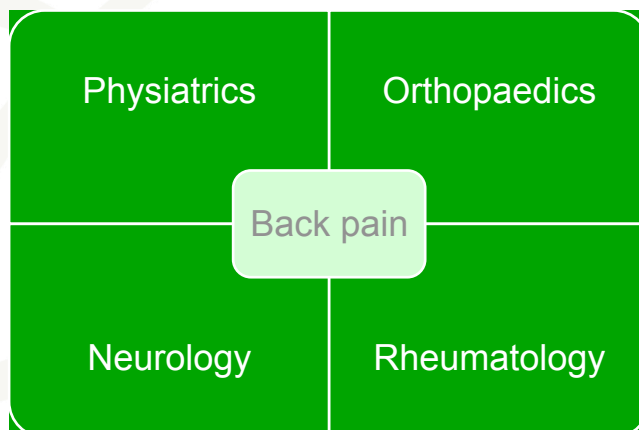


hyvälääkäri.fi

Suorin tie hyvään
hoitoon

www.hyvalääkäri.fi

Intelligent doctor listing



www.hyvalaakari.fi

 **hyvälääkäri.fi**

&

 **ENIRO**

HyväTerveys

www.hyvalaakari.fi

What have we learned?

Planning and building of a website:

- domains: net, org, fi
- conceptual planning, revenue model
- market investigation

- structure- and frame manuscript, layout, logos
- programme management
- competitive bidding and selecting the supplier

- building the service: system design, backend-frontend implementation
- testing, deployment, documentation
- hosting, backup ja technical support

- functional support and further development

www.hyvalääkäri.fi

What have we learned?

New webservice

- If not enough expertise get more
- Use time/expertise for competitive bidding
- Choose supplier who is capable, experienced , flexible and hungry for business

www.hyvalääkäri.fi

What have we learned?

- Concept abroad doesn't automatically work in Finland
- New concept needs to be simple
- Need for Plan B
- Use funding channels
- Listen to your client
- Cooperation is valuable & useful

www.hyvalääkäri.fi

exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvääläkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	

exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvälääkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	

exove

Drupal in brief

One of the market leaders in open source CMS systems

Very active community developing the product

Community-driven, commercial support available from Aquia, but it doesn't dominate the development in any way

Thousands of contributed modules (over 3000 for Drupal 6, the current version), contrib for short

Used widely around the world, in Finland by Nelonen, YLE, A-Lehdet, Suomi24, Nokia Siemens Networks and a lot more

exove

Drupal as a web application platform

Drupal will give you all the basics you need (for most web apps today): user login, user administration, user profiles, fully functional template system and a lot more...

Contrib modules give you more, friends, extensible profiles and a lot more

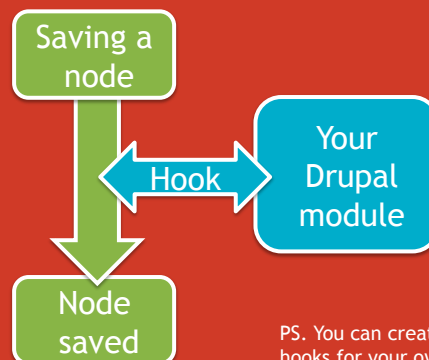
But when you need more, Drupal is made to be easily extensible by creating your own modules

Extension points are hooks and APIs (and toolkits)

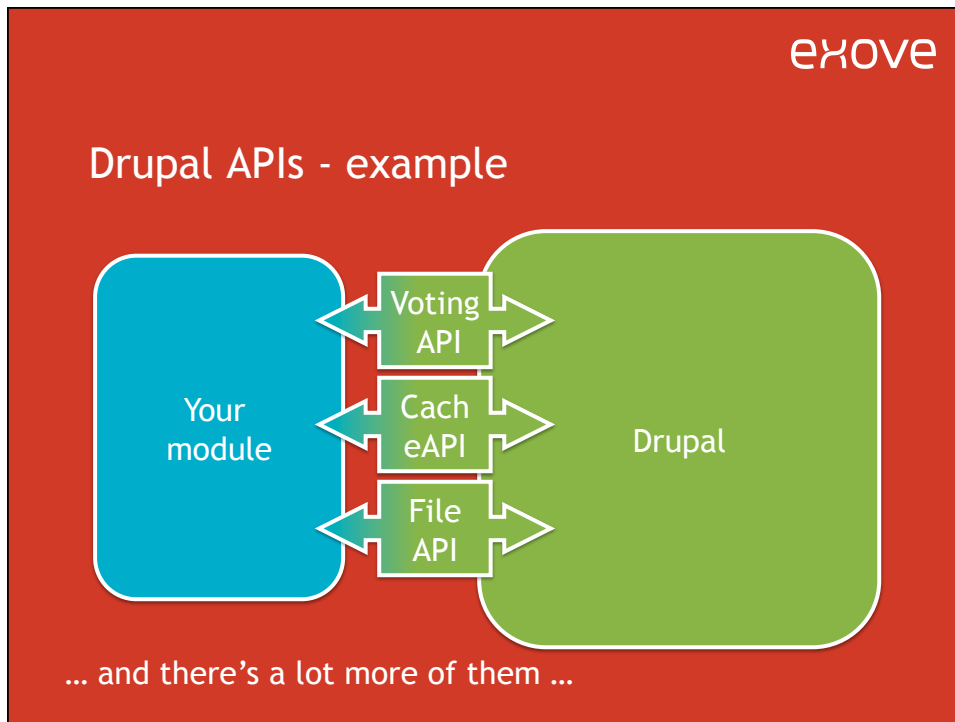
Sometimes you need to touch Drupal's database, but that's not something we'd recommend unless you absolutely know what you're doing

exove

Drupal hooks - example



PS. You can create your own hooks for your own modules for extending them further ...



eXOVE

Challenges in developing on Drupal

Drupal has it's own way of doing stuff - either you obey the "Drupal-way", or use some other platform

Nodes are the main component of any Drupal system, unless you create your own entities

A huge number of nodes (i.e. rows in a single table) might be hazardous for your system's performance

Some of the contrib modules giving you "a free ride" might be incompatible with each other

The screenshot shows a Drupal forum page. The main content is a forum post titled "The Road to Drupal Hell" by Nick Lewis, dated August 7, 2006. The post discusses the Drupal Content Management System's move to an open source layer, referred to as "hell". It includes a FAQ section with questions and answers about Drupal development, such as "What is drupal development hell?", "How much does it cost?", and "Do I have to know PHP, MySQL, or CSS to be damned to drupal development hell?".

The sidebar on the right contains a "User login" form with fields for "Username" and "Password", and a "Log in" button. Below the login form are "Contributor links" including "Community initiatives", "Queues", "My issues", "497 Pending bugs (D7)", "408 Critical issues (D7)", "1742 Patch queue (D7)", "331 Patches to review (D7)", "Performance issues (D7)", "Usability issues (D7)", "Fields in Core issues (D7)", "Play patch bingo!", "Drupal core", "Contributions", "Play bug bingo!", "Drupal core", "Contributions", "Mailing list archives", "Drupal.org webmasters", "Drupal.org server administrators", "Drupal.org CVS applications", and "Web links".

On the far right, there is a vertical URL: <http://drupal.org/node/77487>. The word "exove" is visible in the top right corner of the screenshot area.

The slide is titled "Pros & Cons of using Drupal as a application platform (vs. a framework)". It lists the following points:

- + Good set of base features
- + Great set of contributed modules
- + A lot of integration points (read: hooks and APIs) for extending
- + Big and active community to fix your platform and keep it secure
- Always runs a lot of code (read: it's relatively slow even for easy things)
- Some things are missing in the core and there's not much you can do about it
- Might give you too much restrictions and overhead for really "out there" applications

The word "exove" is visible in the top right corner of the slide.

exove

Example: TUPAS authentication module

TUPAS is a authentication standard used by Finnish banks and issued by Federation of Finnish Financial Services (Finanssialan Keskusliitto)

When using TUPAS authentication, the identity of the end user is established by use of the online banking authentication, which uses disposable passwords

A service provides willing to use TUPAS must integrate to all Finnish banks separately (though the standard is the same for all, they all implement it with some small differences)

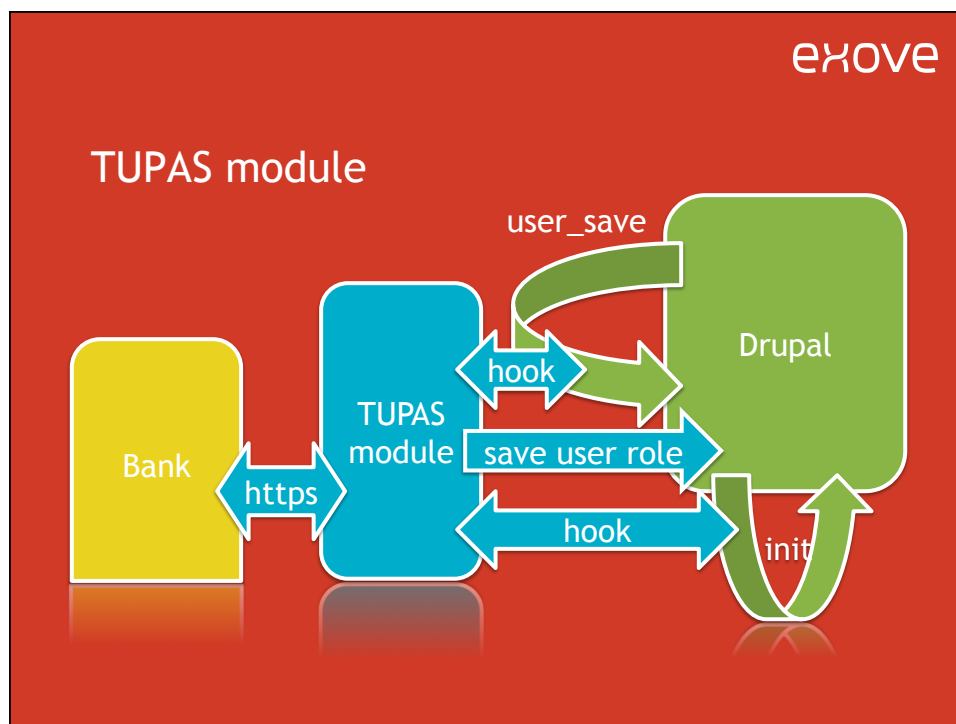
exove

TUPAS module

Exove created a module that authenticates users against all major Finnish banks and released it in Drupal.org for the community

When the authentication is successfully completed, the module will give the authenticated user an extra role in Drupal (a timer can be set for this role)

What the Drupal service does / gives when using this role, is up to the developer using the module, but it could be used for instance for signing agreements or doing some money related decisions that need strong authentication



exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvääläkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	



exove

Agenda

9.00	Opening words	Janne Kalliola, Exove Oy
9.10	CodeIgniter	Janne Kalliola, Exove Oy
9.30	Hyvälääkäri.fi	Pia Koivu, Kvalion Oy
9.50	Break	
10.00	Drupal	Kalle Virta, Exove Oy
10.20	Jaiku	Petteri Koponen, Lifeline Ventures Oy
10.40	Discussion	

exove

Download slides at
<http://www.exove.fi/yritys/seminaarimateriaali/>