Front page

- Contribute menu on the left hand side should contain: “Submit a pattern”, “Post a modeling issue”, “Review a pattern”, “Feedback about the portal”, “Request an account”.
    - “Submit a pattern” leads to a page where all submission types are listed with links to the submission pages (all these are pages corresponding to this http://ontologydesignpatterns.org/wiki/Submissions:ProposeCP for content patterns), and there is a link to the page about pattern types.
    - “Post a modeling issue”, in this page we need to add information on how the modeling issues are handled, i.e. that you create a page and then edit it, that you can provide comments etc.
    - “Review a pattern”, link to http://ontologydesignpatterns.org/wiki/Reviews:PostReview and in this page I would like there to also be a list of pending reviews for the current user, if this is possible? So that if I have been assigned two reviews they will be listed below the current selection options as “My pending reviews” or something like this, to make it easy to reach the patterns you are supposed to review.
    - “Feedback about the portal”, just rename the link to be more clear.
    - “Request an account” moved from help, because you need the account only for contributing, right?
- Navigation menu on the left hand side needs to contain (in this order): “Main page”, “List patterns”, “Pattern types”, “Modeling issues”, “Training”, “Events”. (“Reviews” should be linked from each pattern-type, i.e. from the content pattern page there is a link to the reviews for content patterns etc. “Domains” is mostly a content pattern specific thing, this should be linked from the pattern types that needs it, i.e. content patterns. “Feedback” is a kind of administration thing, where you can check if your feedback has been addressed. Not sure where it should be linked, but feels too specific to have in the main menu.)
    - “List patterns” should link to a page where you can select which pattern type you are interested in, and if you want to see the official catalogue or the submissions (or maybe both?), then you can get to a new page to list only those.
- Help menu: “About ODP”, “What is a pattern?”, “How to post a pattern”, “Training”. (“Request an account” moves to contribute menu.)
    - “About ODP”, “What is a pattern?”, link to the pages currently linked from the front page (not in menu at the moment).
    - “How to post a pattern”, a ne page with some general instructions (like: you need an account, you need to provide information about the pattern) then links to the pages for the different pattern types, and a link to the page explaining the pattern types.
    - “Training” can be duplicated here I think, both in navigation and help.

- Catalogues should contain links to “Content ODPs”, “Re-engineering ODPs”, “Alignment ODPs”, “Logical ODPs”, and “Architechtural ODPs”, and “Beautiful ontologies”. Each of the pattern links should in principle go to the “official catalogue page”, but until we have any patterns in the catalogues, perhaps it is better to link to the submissions page. Or do you think it is better with a step in between, having two links “Official catalogue (0 patterns)” and “Current submissions (42 patterns)” where the numbers are updated automatically? This could be an ok solution?
- Implement you idea of dividing the “menu” with icons in the center as a more structured menu…
- It would be nice to have a nicer look of the news box. Heading of the news should appear first, then the date and who posted it and appear below in much smaller font size, not to take the focus off the news itself.
- Logo… what about some art work similar to our website. Do we have any other pictures we can use? I only know of the lab logo and the map on the front page, but perhaps Valentina has more?

Submission of patterns

In general I think we should distinguish between the submission page of a pattern type and the import/upload/creation page of the pattern. The submission page for Content ODPs is http://ontologydesignpatterns.org/wiki/Submissions:ProposeCP where you have all the information on how to do it and all links you need to do it (to the image upload, to the import page etc.). But right now we sometimes link directly to the import page http://ontologydesignpatterns.org/wiki/Special:ImportProposal which I think should only be accessible from the submission page (above), so that all people access the import from the instructions, and not directly.

- Change the Content ODP submission so that the import is not linked from anywhere but the submission page.
- Add submission pages similar to the one for Content ODPs for the four new pattern types; Logical ODPs, Architectural ODPs, Alignment ODPs, Re-engineering ODPs. In each page, include instructions for submission, and links to image upload and the actual “creation page”.
- For each new type create a “creation page” where the submitter can input the name of the pattern and create the page. I feel that for now, since we do not have any schemas to import information from, the form for editing the pattern information could appear already here, at the same time as you input the name you also give all the other information and link to the picture etc. This feels more natural than to first create an empty page and then edit it, do you agree?
- In addition to the individual forms below, each form should have (at the top so it is not missed) a checkbox or something like this saying “submission to event” and then a drop down menu containing all events that have dates that are not yet passed (or should we even make it “submission deadline” that is not yet passed? And each even has both a date and a “submission deadline”?)

- Is there a way to create property hierarchies? What we would like for the submission templates is a way to have a general set of properties that can then be specialized for each pattern type. For example, in the general template we would have a property named ReusableComponent, this would then have sub-properties such as OWLBuildingBlock for Content ODPs, and Implementation in the case of Re-engineering ODPs. Just in case this is possible I provide below the general template/schema and then the mappings/specializations for each pattern type, implement the general one just as a set op properties and then use the specialized ones in the templates/forms. Assuming that there is a way to create sub-properties there is another thing: in the case where a specific pattern type can directly use the general property, should we then use that property directly of should we anyway create a sub-property (with a very similar name). Consider that the specialized property types might change quite often in the beginning before we have consensus, while the general properties should be quite stable. What is better considering that these changes will then appear after that several pattern have been submitted and described using the old properties?
- I assume that a good practice is to use several instances of a property in case we want to point at several things, is that right? So in the cases where we could want to submit several entries for a property, we should have something like and "add another" button, to add several of the same type. This could be for example for related patterns (rather than listing them in a text string, we would like to add the URIs of several of them, but separately). This could also be applicable to for example CQs, so that we can single out individual CQs rather than having just a string of all of them. If possible, could you try to implement this where you feel it is applicable? (I marked some with *inkl. "add another"* in the tables below.)
- I would like to add a property signifying where the pattern is in the review process as well. When you submit a pattern it is automatically set to the value "submitted", then when a review has been asked for it changes to "review requested", when a reviewer has been assigned it changes to "under review", and when there is some decision from the responsible editor (a conference chair if this is for an event or otherwise the person responsible for the pattern type in the portal) it can either be assigned the value "accepted for event", "rejected for event" (in case it was submitted for an event) or "accepted for certification" or "certification rejected". This property is only applicable to submissions, as soon as the pattern is certified it is no longer applicable.

Content pattern submission

- Use the current set of properties as it is done now, but if possible link them as sub-properties of the general template.
- Add the checkbox and the drop-down list for selecting it as a submission to an event.

General template headings (property name)	**Content ODP template headings (property name)**	Comment
Name	Name	(string)

(PatternName)	(ContentODPName)	
Also known as (PatternAlsoKnownAs)	Also known as (ContentODPAlsoKnownAs)	(string)
Author (PatternAuthor)	Author (ContentODPAuthor)	NEW – inkl. “add another”
Graphical illustration (PatternIllustration)	Image file name (ContentODPIllustration)	A file in the wiki
Domain (PatternDomain)	Domain (ContentODPdomain)	Existing domain in the wiki
Problem (PatternProblem)	Intent (HasIntent), Competency Questions (CoversRequirement)	OBS - 2! – inkl. “add another” on CQs
Solution (PatternSolution)	Solution description (ContentODPDescription)	NEW – text field
Implementation workflow (PatternWorkflow)		N/A for content patterns
Reusable component (PatternReusableComponent)	Reusable OWL building block (OWLBuildingBlock)	Owl-file
Problem example (PatternProblemExample)	Scenario (Scenario)	inkl. “add another”
Solution example (PatternSolutionExample)	Example (CPInstantiationExample)	OWL-file
Consequences (PatternConsequences)	Consequences (HasConsequence)	Text field
Origin (PatternOrigin)	Extracted From (ExtractedFrom), Reengineered From (ReengineeredFrom)	OBS - 2!
Known use (PatternUsage)	Known use (KnownUse)	string
Reference (PatternReference)	Web References (WebReference), Other Reference (Reference)	OBS - 2! – Web ref is URL
Related to (RelatedPattern)	Has Components (HasComponent), Specialization Of (isSpecializationOf), Related CP (RelatedCP)	OBS - 3! – these are URIs to patterns
Validation test (PatternTest)	?	This exists in the annotation schema but not in the portal template? Is this intentional?
URI (PatternURI)	(the URI of the pattern file)	Does this need to have a property defined?
Pattern type (PatternType)	(the category of the pattern page)	
Submitted by (SubmittedBy)	Submitted by (SubmittedBy)	
Submission date	(timestamp?)	Does this need to

(CreationDate ?)		have a property defined?
Sumbitted to event (SubmittedToEvent)	Submitted to event (SubmittedToEvent)	NEW
Status in review process (PatternStatus)	Status in review process (PatternStatus)	NEW – se explanation in text

Re-engineering pattern submission

- Take Boris template, and form, from the testarea and transfer it to the portal, all the properties should be sub-properties of the general template – see left column of the table above (I think the mapping should be straightforward, you can have a look at how the other types of patterns use the general properties (see tables above and below) and try to do the same for Boris template. I know he has more sections, so there will be many cases where 3 or 4 properties are specializations of the same property in the general template.).
- Add the checkbox and drop down menu for submitting to an event, and the status.

Alignment patterns

- Take the general form proposed below, and implement it for alignment patterns. I assume that later Francois wants to perhaps rename the properties but at the moment we can use the general template as it is.

General template headings (property name)	**Alignment ODP template headings (property name)**
Name (PatternName)	Name (AlignmentODPName)
Also known as (PatternAlsoKnownAs)	Also known as (AlignmentlODPAlsoKnownAs)
Author (PatternAuthor)	Author (AlignmentODPAuthor)
Graphical illustration (PatternIllustration)	Illustration of alignment (AlignmentODPIllustration)
Domain (PatternDomain)	Domain (if applicable) (AlignmentODPDomain)
Problem (PatternProblem)	Alignment problem addressed (AlignmentODPProblem)
Solution (PatternSolution)	Alignment solution (AlignmentODPSolution)
Implementation workflow (PatternWorkflow)	Alignment workflow (AlignmentODPWorkflow)
Reusable component (PatternReusableComponent)	Reusable component (AlignmentODPComponent)
Problem example (PatternProblemExample)	Example scenario (AlignmentODPScenario)
Solution example (PatternSolutionExample)	Solution example (AlignmentODPExample)
Consequences (PatternConsequences)	Consequences

	(AlignmentODPConsequence)
Origin (PatternOrigin)	Origin (AlignmentODPOrigin)
Known use (PatternUsage)	Known use (AlignmentODPKnownUse)
Reference (PatternReference)	Reference (AlignmentODPReference)
Related to (RelatedPattern)	Related ODP (AlignmentODPRelated)
Validation test (PatternTest)	Test (AlignmentODPTest)
URI (PatternURI)	(the URI of the pattern in the portal)
Pattern type (PatternType)	(the category of the patten page)
Submitted by (SubmittedBy)	Submitted by (SubmittedBy)
Submission date (CreationDate ?)	(timestamp?)
Sumbitted to event (SubmittedToEvent)	Submitted to event (SubmittedToEvent)
Status in review process (PatternStatus)	Status in review process (PatternStatus)

Alignment ODP template headings (property name)	**(?) - Help text**
Name (AlignmentODPName)	An illustrative and unique name
Also known as (AlignmentlODPAlsoKnownAs)	Other names for this pattern
Author (AlignmentODPAuthor)	The author(s) of this pattern (if dfferent from the user submitting it)
Illustration of alignment (AlignmentODPIllustration)	A graphical illustration of the pattern structure
Domain (if applicable) (AlignmentODPDomain)	If this pattern is tailored to a specific domain or type of applications, add the domain here
Alignment problem addressed (AlignmentODPProblem)	What is the problem that this pattern tries to solve?
Alignment solution (AlignmentODPSolution)	How is the problem solved by this pattern?
Alignment workflow (AlignmentODPWorkflow)	How should this pattern be applied, what are the steps?
Reusable component (AlignmentODPComponent)	A reusable component, for example as a formal description of the alignment
Example scenario (AlignmentODPScenario)	An example of a problem scenario where this pattern is applicable
Solution example (AlignmentODPExample)	An example solution to the scenario
Consequences (AlignmentODPConsequence)	What are the consequences of using this alignment pattern? Are there limitations?
Origin (AlignmentODPOrigin)	Does this pattern originate in some other work, or some other pattern?
Known use (AlignmentODPKnownUse)	What existing ontologies have used this type of alignments?
Reference (AlignmentODPReference)	References to papers, web-pages or other resources.
Related ODP (AlignmentODPRelated)	Related patterns, such as specializations of this pattern, components or combinations etc. What patterns are suitable to combine with this pattern?

Test (AlignmentODPTest)	A test case that can be applied to check that the pattern was correctly instantiated.
(the URI of the pattern in the portal)	
(the category of the patten page)	
Submitted by (SubmittedBy)	
(timestamp?)	
Submitted to event (SubmittedToEvent)	
Status in review process (PatternStatus)	

Logical patterns

General template headings (property name)	**Logical ODP template headings (property name)**	Comment
Name (PatternName)	Name (LogicalODPName)	
Also known as (PatternAlsoKnownAs)	Also known as (LogicalODPAlsoKnownAs)	
Author (PatternAuthor)	Author (LogicalODPAuthor)	
Graphical illustration (PatternIllustration)	Illustration of structure (LogicalODPIllustration)	
Domain (PatternDomain)		Not applicable
Problem (PatternProblem)	Motivation (LogicalODPMotivation), Aim (LogicalODPAim)	OBS - 2!
Solution (PatternSolution)	Solution description (LogicalODPDescription), Elements (LogicalODPElements)	OBS - 2!
Implementation workflow (PatternWorkflow)	Implementation (LogicalODPWorkflow)	
Reusable component (PatternReusableComponent)	Reusable component (LogicalODPComponent), Component type (LogicalODPComponentDescription)	OBS - 2!
Problem example (PatternProblemExample)	Example scenario (LogicalODPScenario)	
Solution example (PatternSolutionExample)	Sample (LogicalODPSample)	
Consequences (PatternConsequences)	Result and consequences (LogicalODPConsequence)	
Origin (PatternOrigin)	Origin (LogicalODPOrigin)	
Known use (PatternUsage)	Known use (LogicalODPKnownUse)	
Reference (PatternReference)	Reference (LogicalODPReference)	
Related to	Related ODP (LogicalODPRelated),	OBS - 2!

(RelatedPattern)	Used in Combination With (LogicalODPCombination)
Validation test (PatternTest)	Test (LogicalODPTest)
URI (PatternURI)	(the URI of the pattern in the portal)
Pattern type (PatternType)	(the category of the patten page)
Submitted by (SubmittedBy)	Submitted by (SubmittedBy)
Submission date (CreationDate ?)	(timestamp?)
Sumbitted to event (SubmittedToEvent)	Submitted to event (SubmittedToEvent)
Status in review process (PatternStatus)	Status in review process (PatternStatus)

Logical ODP template headings (property name)	**(?) - Help text**
Name (LogicalODPName)	An illustrative and unique name
Also known as (LogicalODPAlsoKnownAs)	Other names for this pattern
Author (LogicalODPAuthor)	The author(s) of this pattern (if different from the user submitting it)
Illustration of structure (LogicalODPIllustration)	A graphical illustration of the pattern structure
Motivation (LogicalODPMotivation),	The motivation why this pattern is needed
Aim (LogicalODPAim)	The aim, i.e. what problem it intends to solve
Solution description (LogicalODPDescription),	A description of the overall solution
Elements (LogicalODPElements)	The elements of the solution
Implementation (LogicalODPWorkflow)	A description, workflow, showing how this pattern should be applied
Reusable component (LogicalODPComponent),	A reusable component, either as an OWL-file or in some other formal representation
Component type (LogicalODPComponentDescription)	A description of the reusable component, its format and how it can be applied
Example scenario (LogicalODPScenario)	An example of a problem scenario where this pattern would be applicable
Sample (LogicalODPSample)	A sample solution for the example scenario
Result and consequences (LogicalODPConsequence)	What are the effects of applying this pattern? What is the result?
Origin (LogicalODPOrigin)	Does this pattern originate in some other work, or some other pattern?

Known use (LogicalODPKnownUse)	In what ontologies has this pattern been applied?
Reference (LogicalODPReference)	References to papers, web pages or other resources.
Related ODP (LogicalODPRelated),	Related patterns, such as specializations of this pattern, components or combinations etc.
Used in Combination With (LogicalODPCombination)	Pattern that this pattern has been used in combination with.
Test (LogicalODPTest)	A test case that can be applied to check that the pattern was correctly instantiated.
(the URI of the pattern in the portal)	
(the category of the patten page)	
Submitted by (SubmittedBy)	
(timestamp?)	
Submitted to event (SubmittedToEvent)	
Status in review process (PatternStatus)	

Architecture patterns

General template headings (property name)	**Architectural ODP template headings (property name)**	Comment
Name (PatternName)	Name (ArchitecturalODPName)	
Also known as (PatternAlsoKnownAs)	Also known as (ArchitecturalODPAlsoKnownAs)	
Author (PatternAuthor)	Author (ArchitecturalODPAuthor)	
Graphical illustration (PatternIllustration)	Illustration of structure (ArchitecturalODPIllustration)	
Domain (PatternDomain)	Domain (if applicable) (ArchitecturalODPDomain)	
Problem (PatternProblem)	Problem description (ArchitecturalODPProblem)	
Solution (PatternSolution)	Solution description (ArchitecturalODPSolution)	
Implementation workflow (PatternWorkflow)	Implementation workflow (ArchitecturalODPWorkflow)	
Reusable component (PatternReusableComponent)	Reusable component (ArchitecturalODPComponent)	
Problem example (PatternProblemExample)	Example scenario (ArchitecturalODPScenario)	
Solution example (PatternSolutionExample)	Solution example (ArchitecturalODPExample)	
Consequences (PatternConsequences)	Consequences (ArchitecturalODPConsequence)	
Origin (PatternOrigin)	Origin (ArchitecturalODPOrigin)	
Known use (PatternUsage)	Known use (ArchitecturalODPKnownUse)	
Reference (PatternReference)	Reference (ArchitecturalODPReference)	

Related to (RelatedPattern)	Related ODP (ArchitecturalODPRelated)	
Validation test (PatternTest)		Not applicable
URI (PatternURI)	(the URI of the pattern in the portal)	
Pattern type (PatternType)	(the category of the patten page)	
Submitted by (SubmittedBy)	Submitted by (SubmittedBy)	
Submission date (CreationDate ?)	(timestamp?)	
Sumbitted to event (SubmittedToEvent)	Submitted to event (SubmittedToEvent)	
Status in review process (PatternStatus)	Status in review process (PatternStatus)	

Architectural ODP template headings (property name)	**(?) - Help text**
Name (ArchitecturalODPName)	An illustrative and unique name
Also known as (ArchitecturalODPAlsoKnownAs)	Other names for this pattern
Author (ArchitecturalODPAuthor)	The author(s) of this pattern (if dfferent from the user submitting it)
Illustration of structure (ArchitecturalODPIllustration)	A graphical illustration of the pattern structure
Domain (if applicable) (ArchitecturalODPDomain)	If this pattern is tailored to a specific domain or type of applications, add the domain here
Problem description (ArchitecturalODPProblem)	What is the problem that this pattern tries to solve?
Solution description (ArchitecturalODPSolution)	How is the problem solved by this pattern?
Implementation workflow (ArchitecturalODPWorkflow)	How should this pattern be applied, what are the steps?
Reusable component (ArchitecturalODPComponent)	A reusable component, for example an OWL-file providing a "code skeleton" for realizing this architecture
Example scenario (ArchitecturalODPScenario)	An example of a problem scenario where this pattern is applicable
Solution example (ArchitecturalODPExample)	An example solution to the scenario
Consequences (ArchitecturalODPConsequence)	What are the consequences of using this architecture pattern? Are there limitations?
Origin (ArchitecturalODPOrigin)	Does this pattern originate in some other work, or some other pattern?
Known use (ArchitecturalODPKnownUse)	What existing ontologies display good examples of this architecture?
Reference (ArchitecturalODPReference)	References to papers, web-pages or other resources.
Related ODP (ArchitecturalODPRelated)	Related patterns, such as specializations of this pattern, components or combinations etc. What

	patterns are suitable to combine with this pattern?
(the URI of the pattern in the portal)	
(the category of the patten page)	
Submitted by (SubmittedBy)	
(timestamp?)	
Submitted to event (SubmittedToEvent)	
Status in review process (PatternStatus)	

WOP page

- In the menu to the left put an entry for “Submitted patterns”, which is a page where you query for pages that are patterns and are submitted to the event WOP2009.