

Applied Workflows in Geodise

e-Science Workflow Services Workshop
Edinburgh (Dec 3rd – 5th 2003)

Dec 4th 2003
Prof Simon Cox

Computational Engineering & Design Group
School of Engineering Sciences
University of Southampton

Grid Enabled Optimisation and Design Search for Engineering (GEODISE)

Southampton, Oxford and Manchester

Simon Cox- *Grid/ W3C Technologies
and High Performance Computing
Global Grid Forum Apps Working Group*

Andy Keane- *Director of Rolls Royce/
BAE Systems University Technology
Partnership in Design Search and
Optimisation*

Mike Giles- *Director of Rolls Royce
University Technology Centre for
Computational Fluid Dynamics*

Carole Goble- *Ontologies and DARPA
Agent Markup Language (DAML) /
Ontology Inference Language (OIL)*

Nigel Shadbolt- *Director of Advanced
Knowledge Technologies (AKT) IRC*

BAE SYSTEMS- *Engineering*

Rolls-Royce- *Engineering*

Fluent- *Computational Fluid Dynamics*

Microsoft- *Software/ Web Services*

Intel- *Hardware*

Compusys- *Systems Integration*

Epistemics- *Knowledge Technologies*

Condor- *Grid Middleware*



University
of Southampton

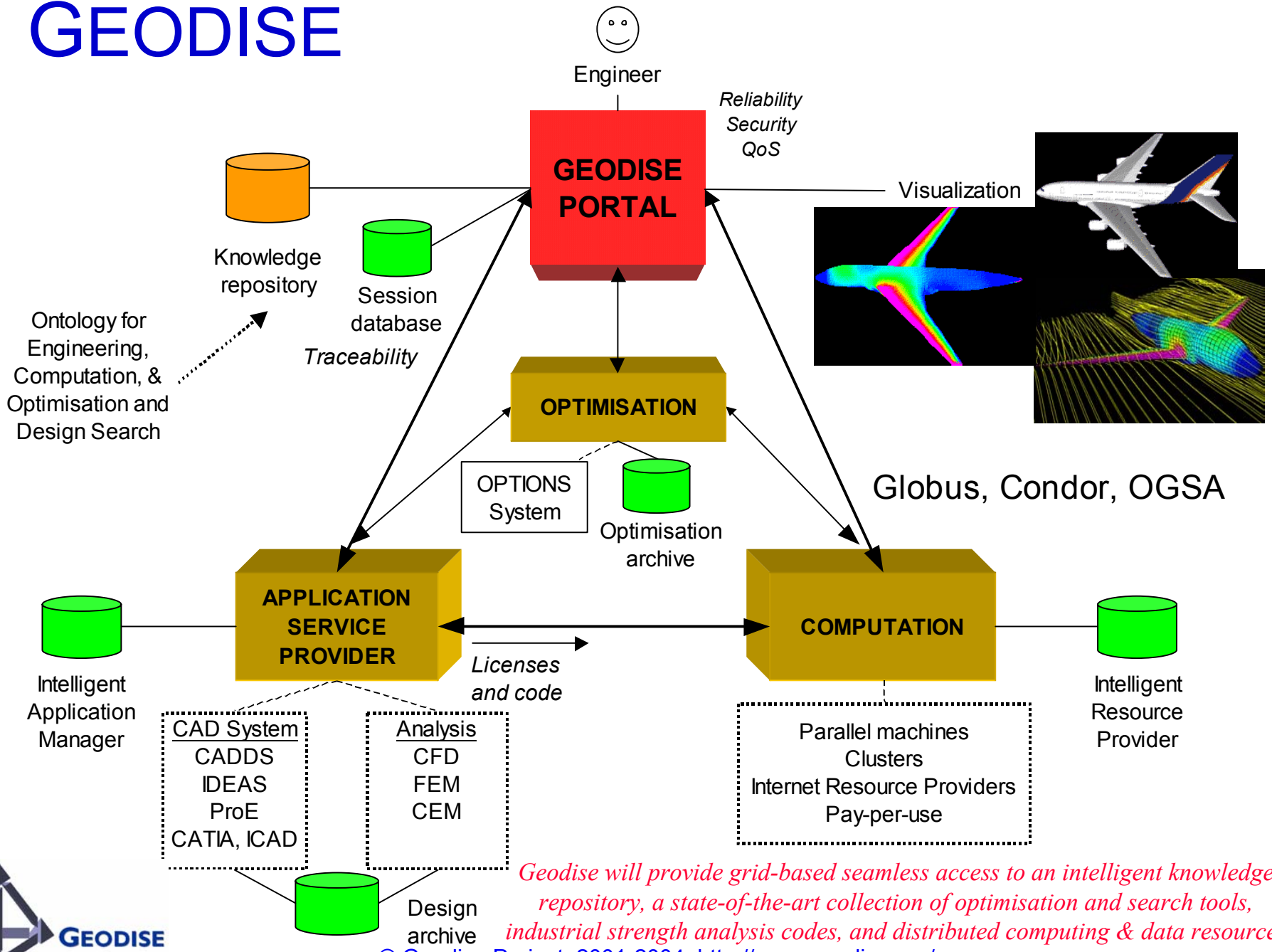


THE UNIVERSITY
of MANCHESTER

The GEODISE Team ...

- Richard Boardman
- Sergio Campobasso
- Liming Chen
- Mike Chrystall
- Trevor Cooper-Chadwick
- Simon Cox
- Mihai Duta
- Clive Emberey
- Hakki Eres
- Matt Fairman
- Mike Giles
- Carole Goble
- Ian Hartney
- Tracey Hunt
- Zhuoan Jiao
- Andy Keane
- Marc Molinari
- Graeme Pound
- Colin Puleston
- Nicola Reader
- Angus Roberts
- Mark Scott
- Nigel Shadbolt
- Wenbin Song
- Paul Smart
- Barry Tao
- Jasmin Wason
- Fenglian Xu
- Gang “Luke” Xue

GEODISE



Geodise will provide grid-based seamless access to an intelligent knowledge repository, a state-of-the-art collection of optimisation and search tools, industrial strength analysis codes, and distributed computing & data resources

A few of my favourite things to do with workflows

- Create
- Retrieve
- Cut 'n' Shut
- Configure
- Execute
- Monitor
- Share
- Steer
- Dynamically modify

When is a script not a script?

... when it's a “workflow”

Scripting languages

Why use scripting languages?

- Flexibility
- High-level functionality
- Quick application development
- Extend the user's existing PSE

Example Script

```
hostname = 'pacifica.iridis.soton.ac.uk'
jobmanager = [hostname, '/jobmanager-fork']
rsl = '&(executable="/bin/date") (stdout="remote.txt") '

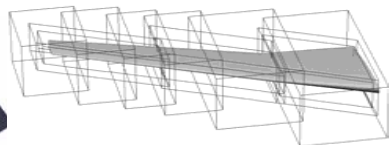
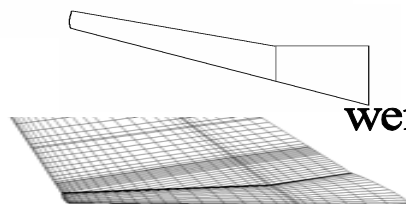
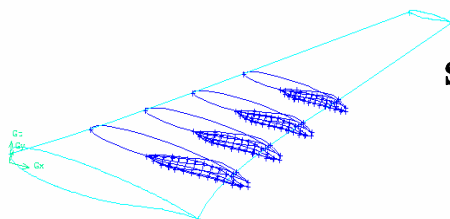
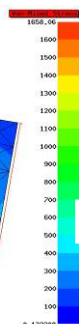
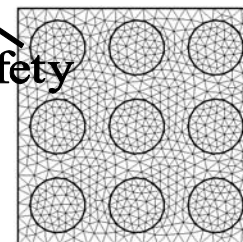
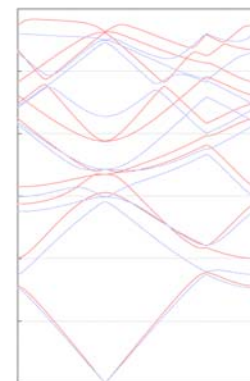
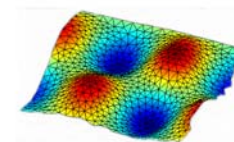
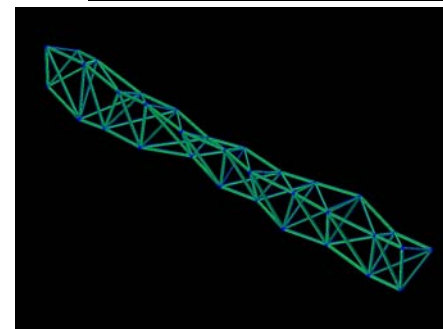
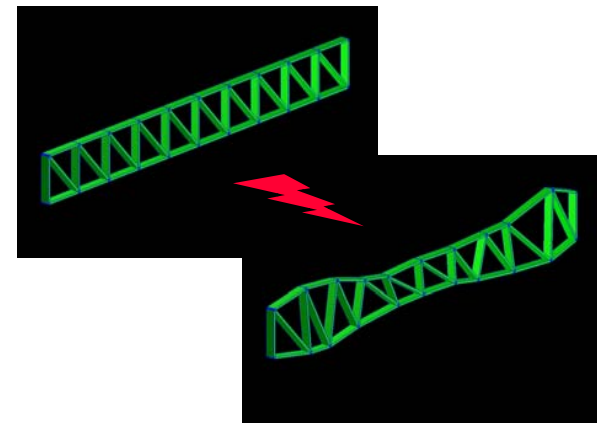
%Create a proxy certificate
gd_createproxy

%Submitting a globus job and returning handle
handle = gd_jobsubmit(rsl, jobmanager)

%Polling the job
gd_jobpoll(handle)

%Getting the standard output
gd_getfile(hostname, 'remote.txt', 'local.txt');

%Print the output to screen
type('local.txt')
```

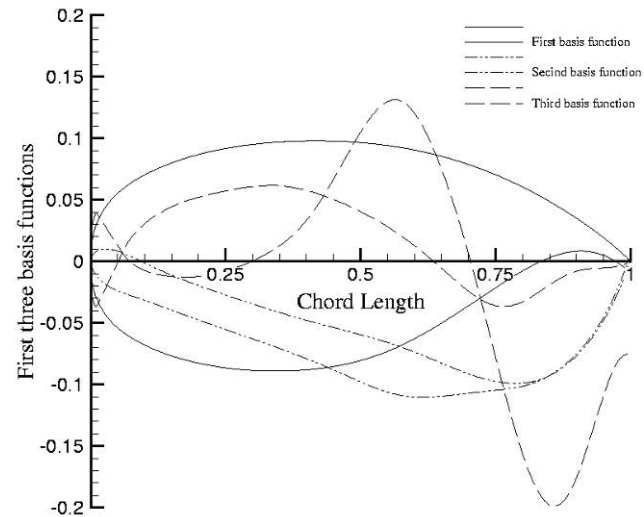



Aerodynamic Shape Optimisation using CFD

- Integration of CAD, mesh generation, and solver
 - Direct API access to CAD models
 - third-party standards based data exchange
- Robust mesh generation
 - Automatic mesh generation
 - Control over mesh properties
- Multi-fidelity models
 - Euler solver
 - Navier Stokes solver

Orthogonal Basis Function for Airfoil Design

- Parameterisation methods for airfoil:
 - mathematical functions
 - empirical basis functions
 - control points based curve fitting
- Orthogonal basis functions
 - unique mapping from parameter space to design space
 - for preliminary wing design
 - fewer number of design variables
 - different set of basis functions for different design task



Robinson, G.M. and Keane, A.J., "Concise Orthogonal Representation of Supercritical Airfoils", Journal of Aircraft 38(3) (2001) 580-583.

Optimisation Workflow and Results

- Optimisation strategy:

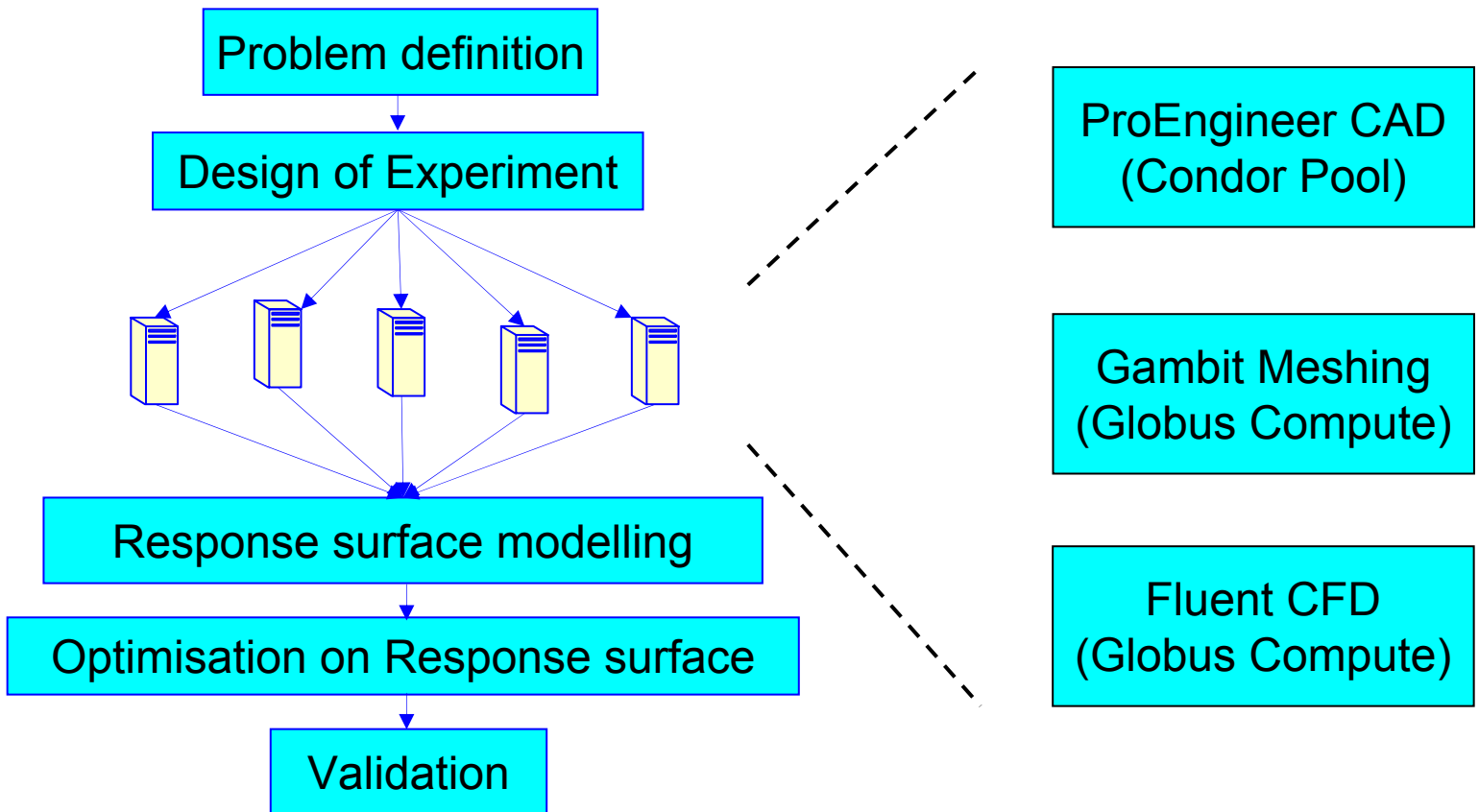
- Single optimisation method proved to be inefficient for practical problems, more complex strategies are required
- Two-stage hybrid approach (Genetic algorithm + Gradient search)

- Surrogate modelling

- CFD runs on complex configurations is too expensive (24hrs/run)
- Surrogate modelling methods
 - Polynomial curve fitting
 - Stochastic method (DACE or Kriging)
 - Neural network
- High-dimensional design space

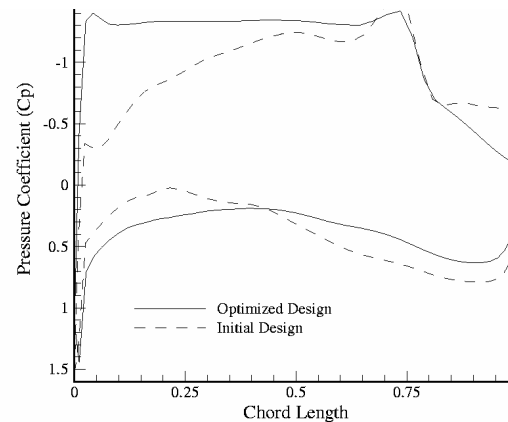
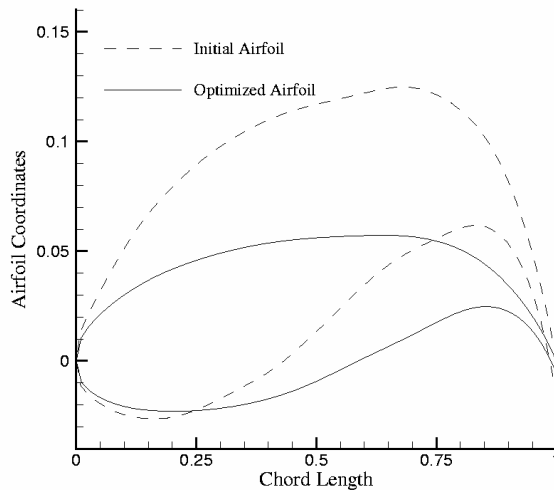
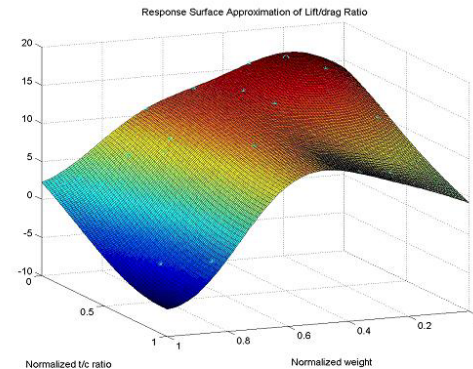
- Combining response surface modelling (RSM)
/two-stage approach

Workflow for aerodynamic shape optimisation using CAD, Gambit, and Fluent



Response surface model and two-stage hybrid search using GA/Local tuning

- Response surface model
- two-stage hybrid search methods
- Comparison of Airfoil shape and pressure distribution



Engine Nacelle Optimisation

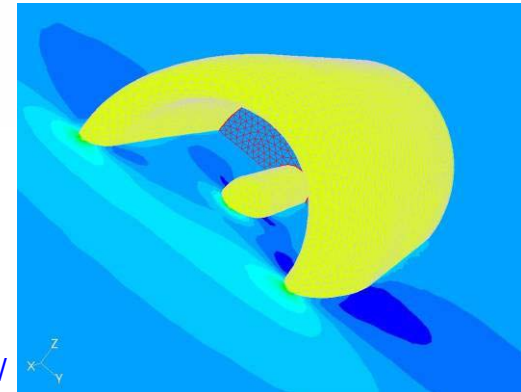
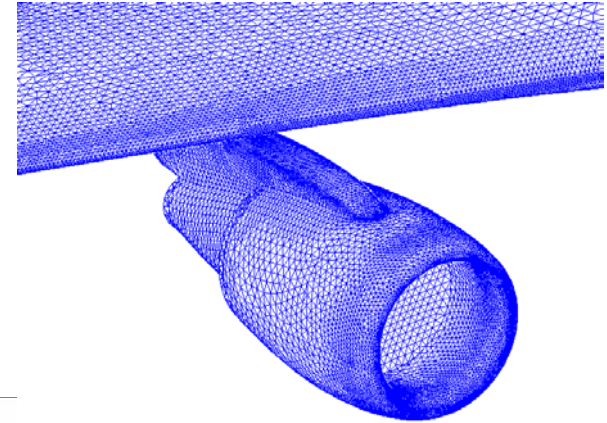
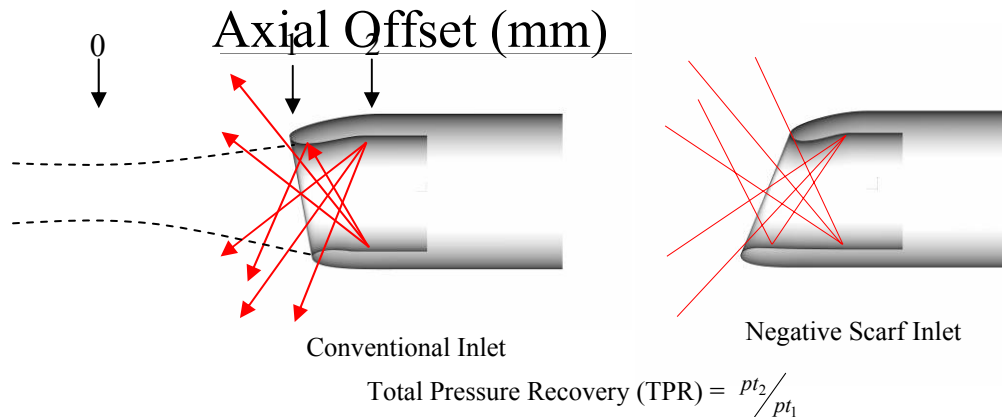
(problem definition)

Assumption: Noise radiated to ground reduces with increasing scarf angle

Objective function:

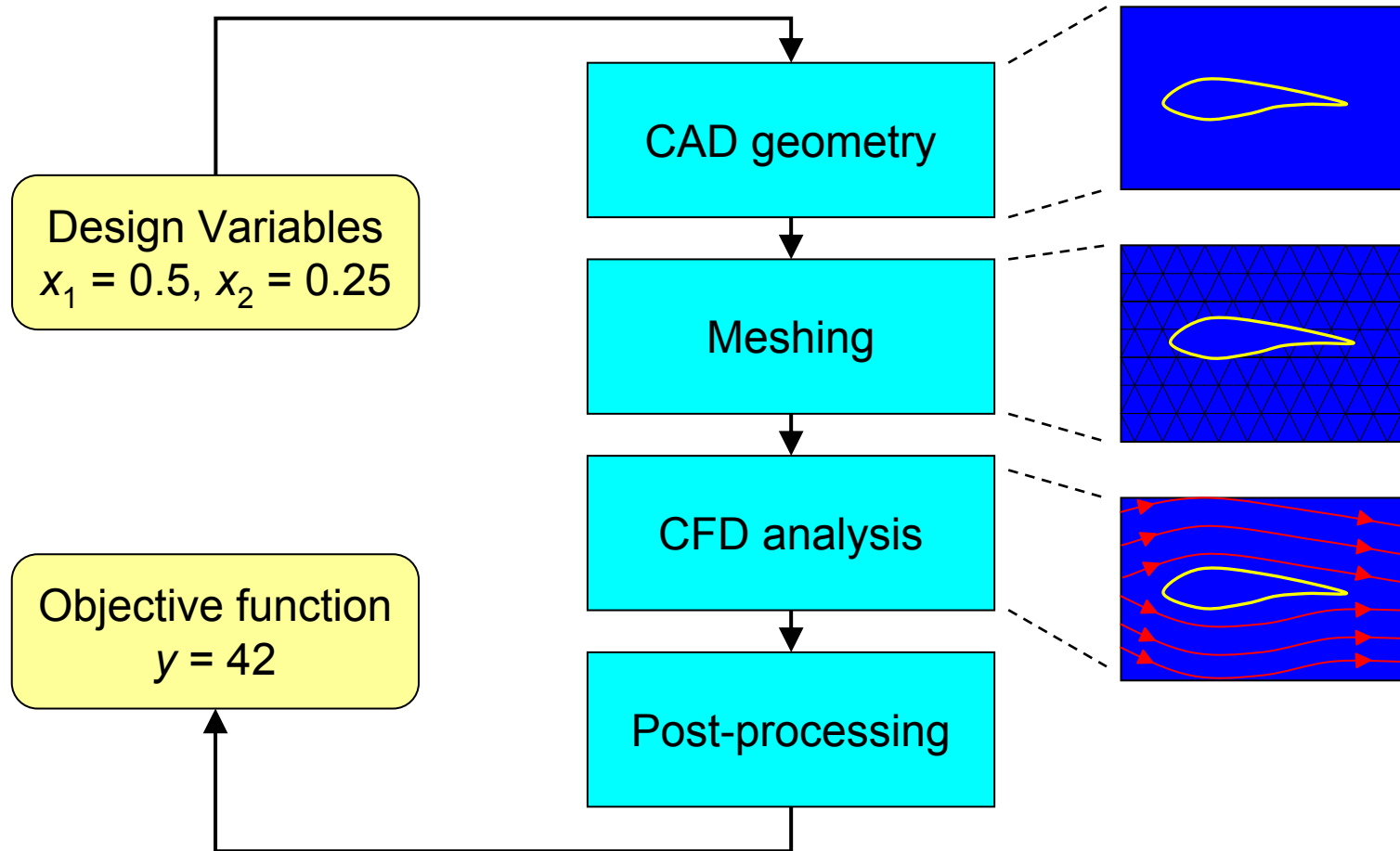
Total Pressure Recovery (pt_2/pt_1)

Design variables: Scarf Angle (degrees)





Defining the Objective Function

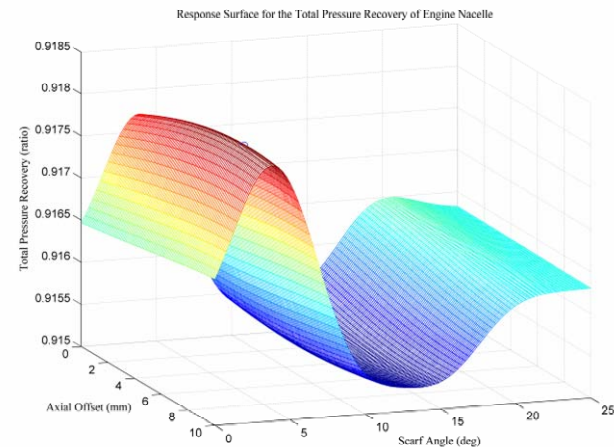
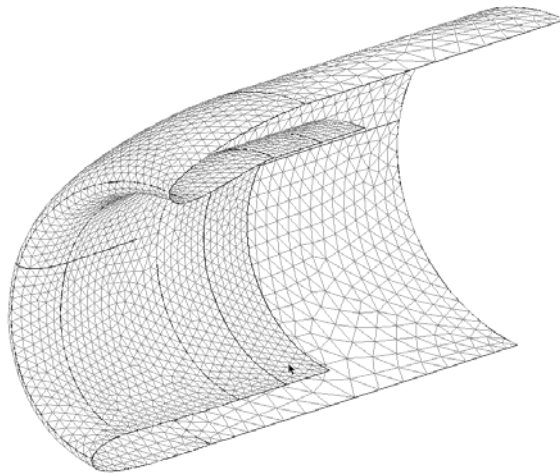


***“Bigger workflows are made from little workflows,
Little workflows are made from littler workflows,
And so on...”***

Engine Nacelle Optimisation (3D)

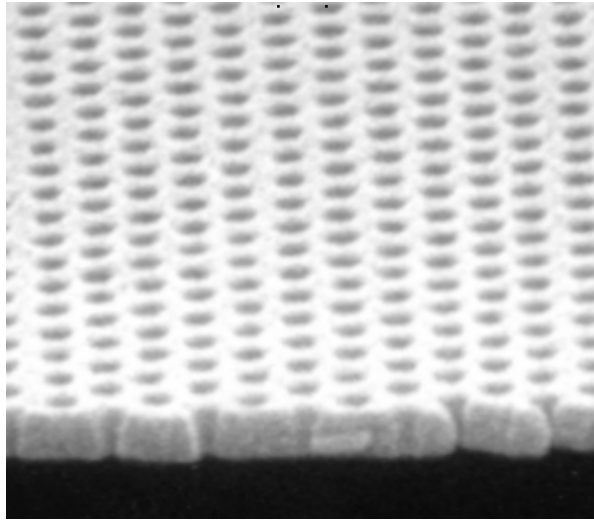
(some results)

- Typical unstructured mesh used in the problem (left)
- Response surface model built for two design variables (right)
- The effect of other geometry parameters need to be investigated



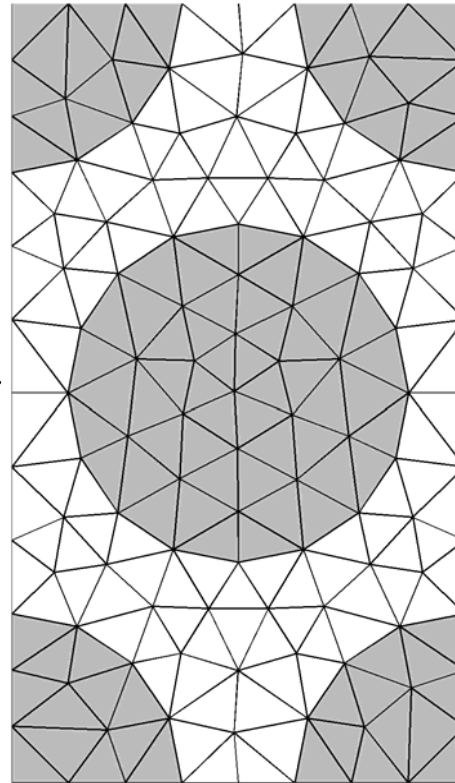
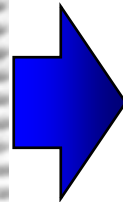
Photonic Device Modelling

pitch=300nm

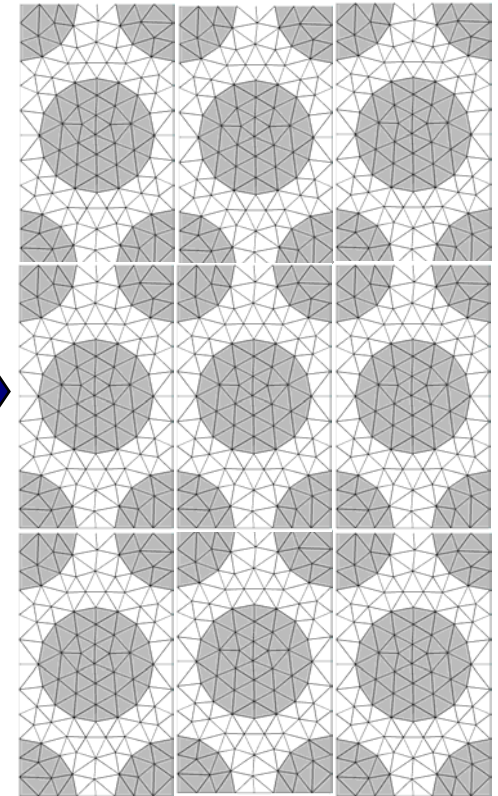
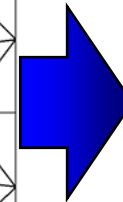


*Bridge waveguide structure courtesy of
Martin Charlton, Southampton
Microelectronics Research Group.*

REAL-THING
(photo)



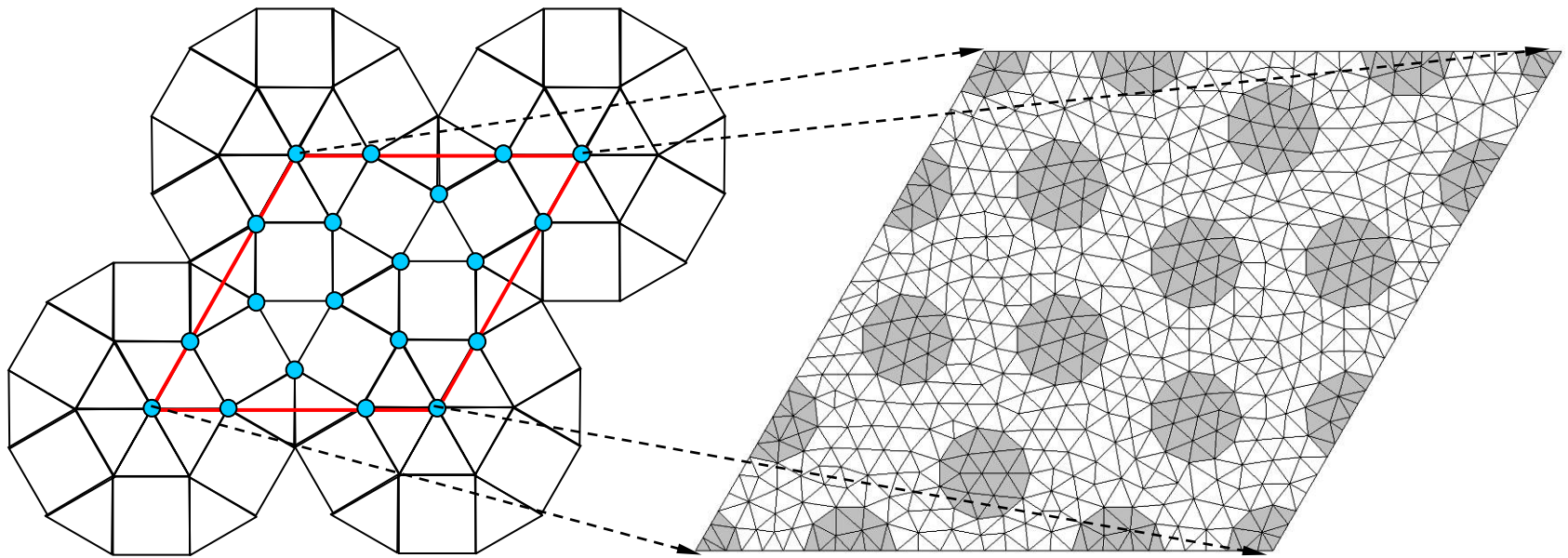
UNIT-CELL



PERIODICALLY
TILED UNIT-CELLS

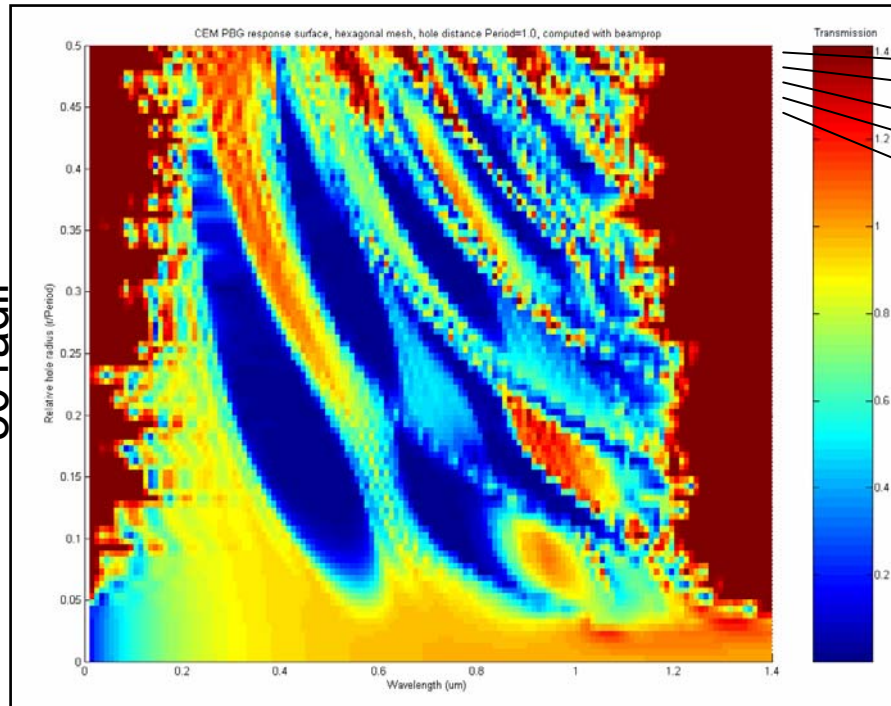
12-fold Symmetric Quasicrystals

- Based on tiling of dodecagons composed of squares and equilateral triangles
- Possesses 12 fold rotational symmetry
- Leads to a highly homogeneous band gap



CEM Simulation Results

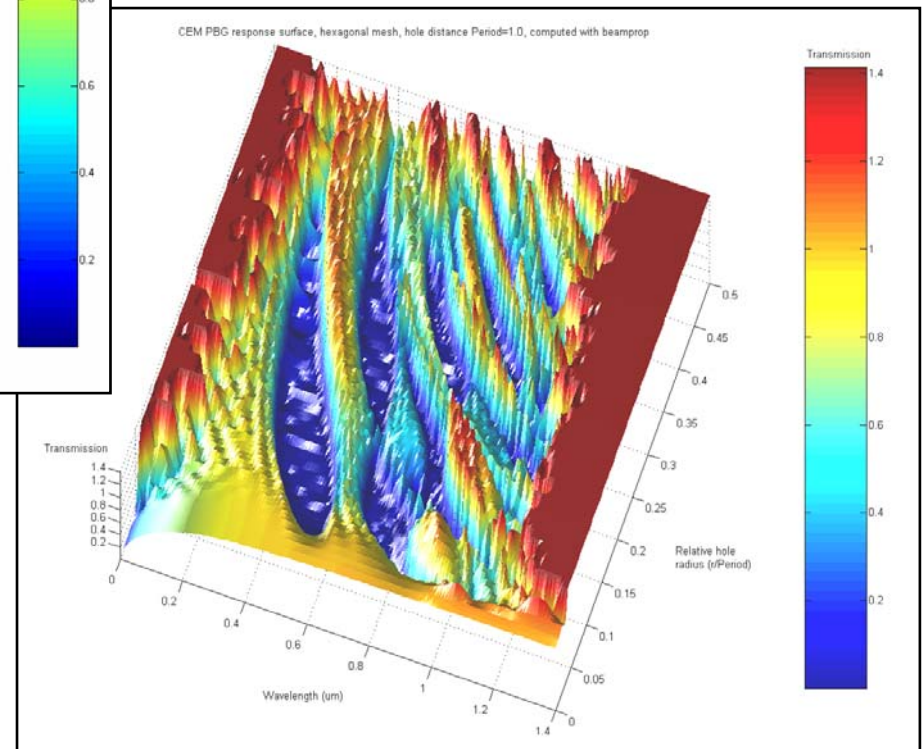
50 radii



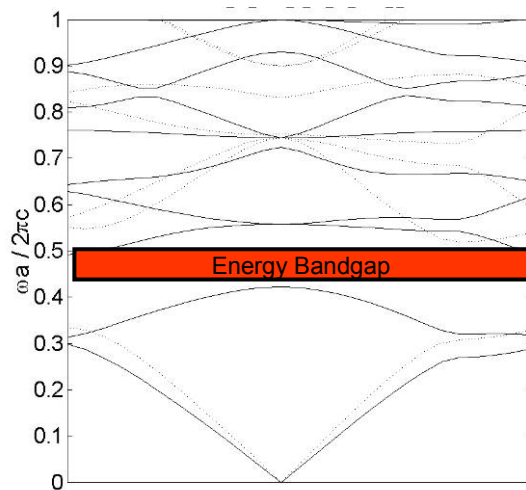
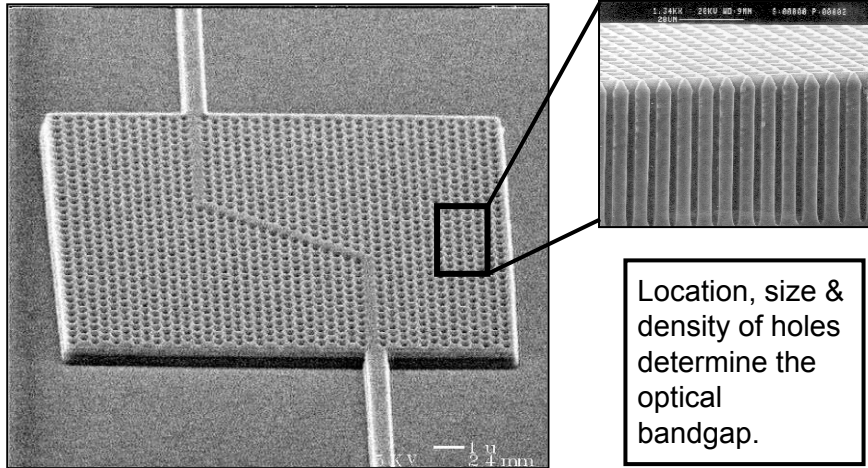
CompResource.1
CompResource.2
CompResource.3
CompResource.4
CompResource.1
...

50 frequencies

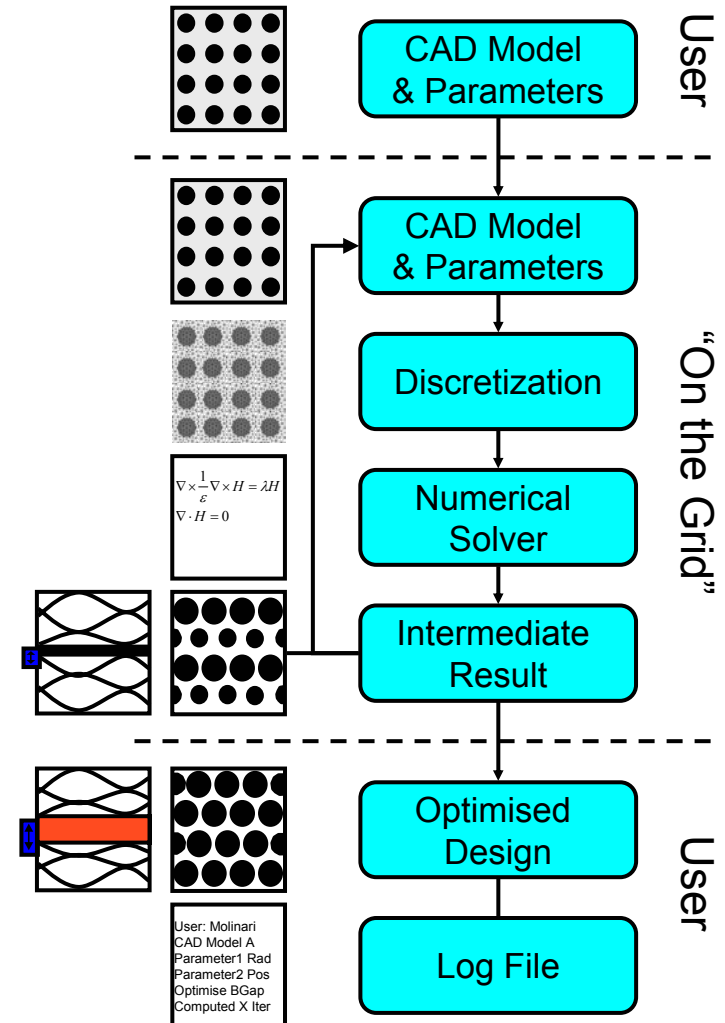
Photonic Crystal Response Surface /
Photonic Band Gap Map



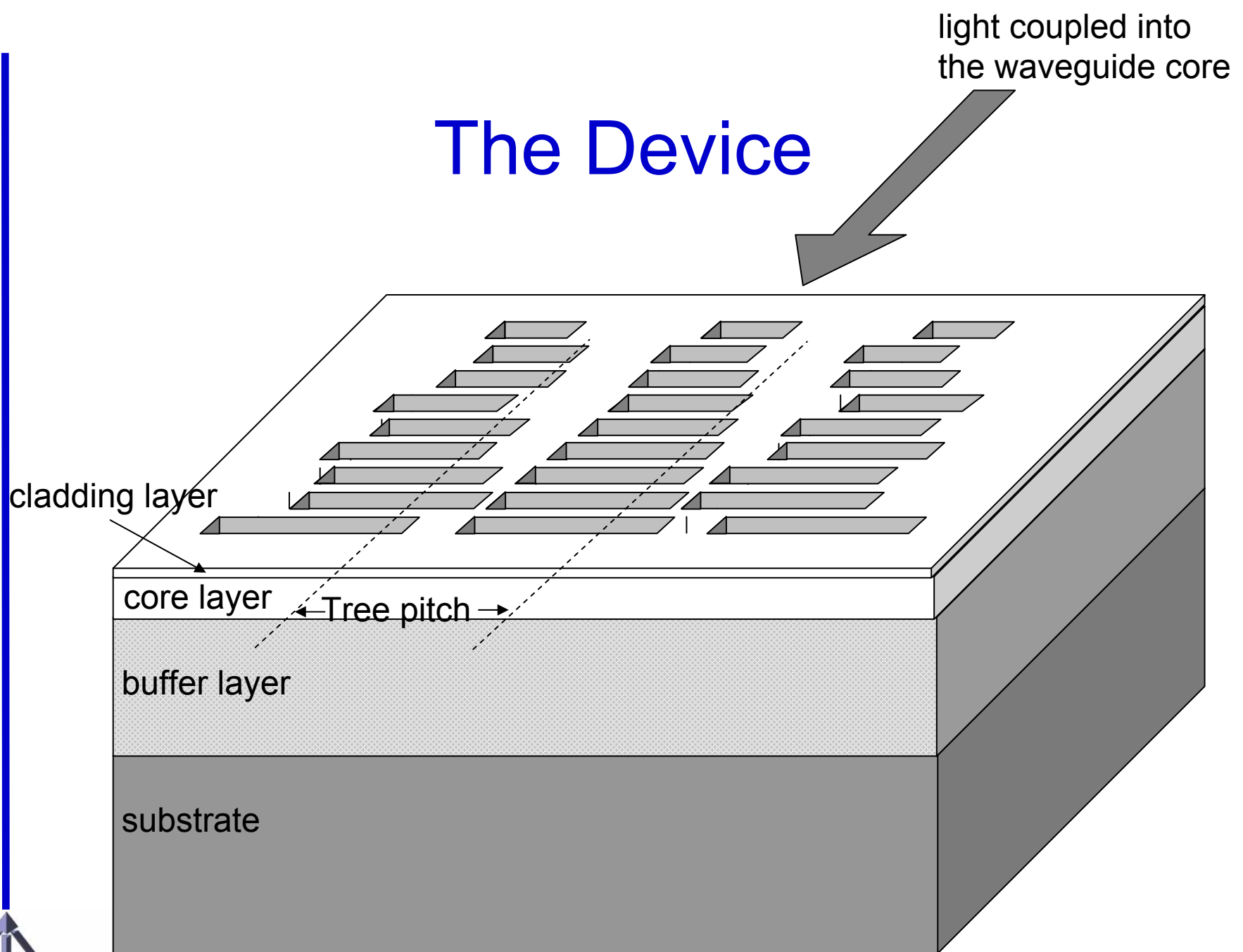
Photonic Crystal - Optimisation



The aim of this design optimisation process is to find a configuration which maximises this bandgap and minimizes energy loss.



The Device



The Script

[geometry_optimise_EDIT.m](#)

Scripting languages

Why use scripting languages?

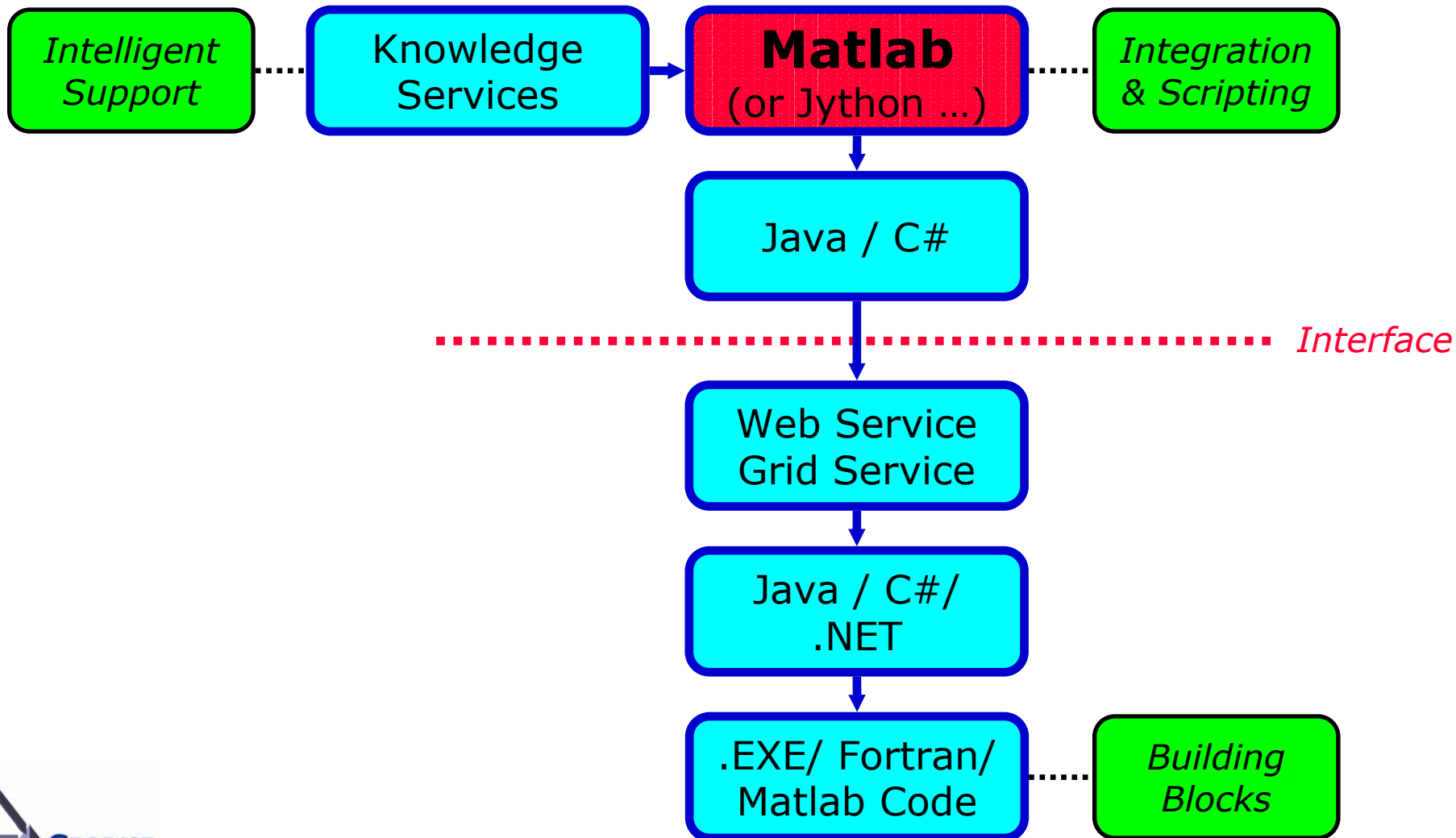
- Flexibility
- High-level functionality
- Quick application development
- Extend the user's existing PSE

Matlab Grid-Enabled Scripting Environment

- Motivations:
 - ❖ Flexible, transparent access to computational resources
 - ❖ Easy to use for engineers (and in widespread use)
- Our Approach
 - ❖ Matlab chosen as the hosting environment
 - ◆ Extend the user's existing PSE
 - ◆ High-level functionality
 - ◆ Quick application development
 - ◆ ... is our execution/ enactment engine too
 - ❖ Computational resources exposed in the form of Matlab functions
 - ◆ Job submission to Globus server using Java Cog
 - ◆ Job submission to Condor pool via Web services interface
 - ❖ Integration of CAD, Mesh generation, and Fluent solver via the use of intermediate data format, often standard-based, or package-neutral
 - ❖ Hybrid search strategies to make the best use of different search methods
- Can also use Python, Jython, etc.



Geodise Architecture

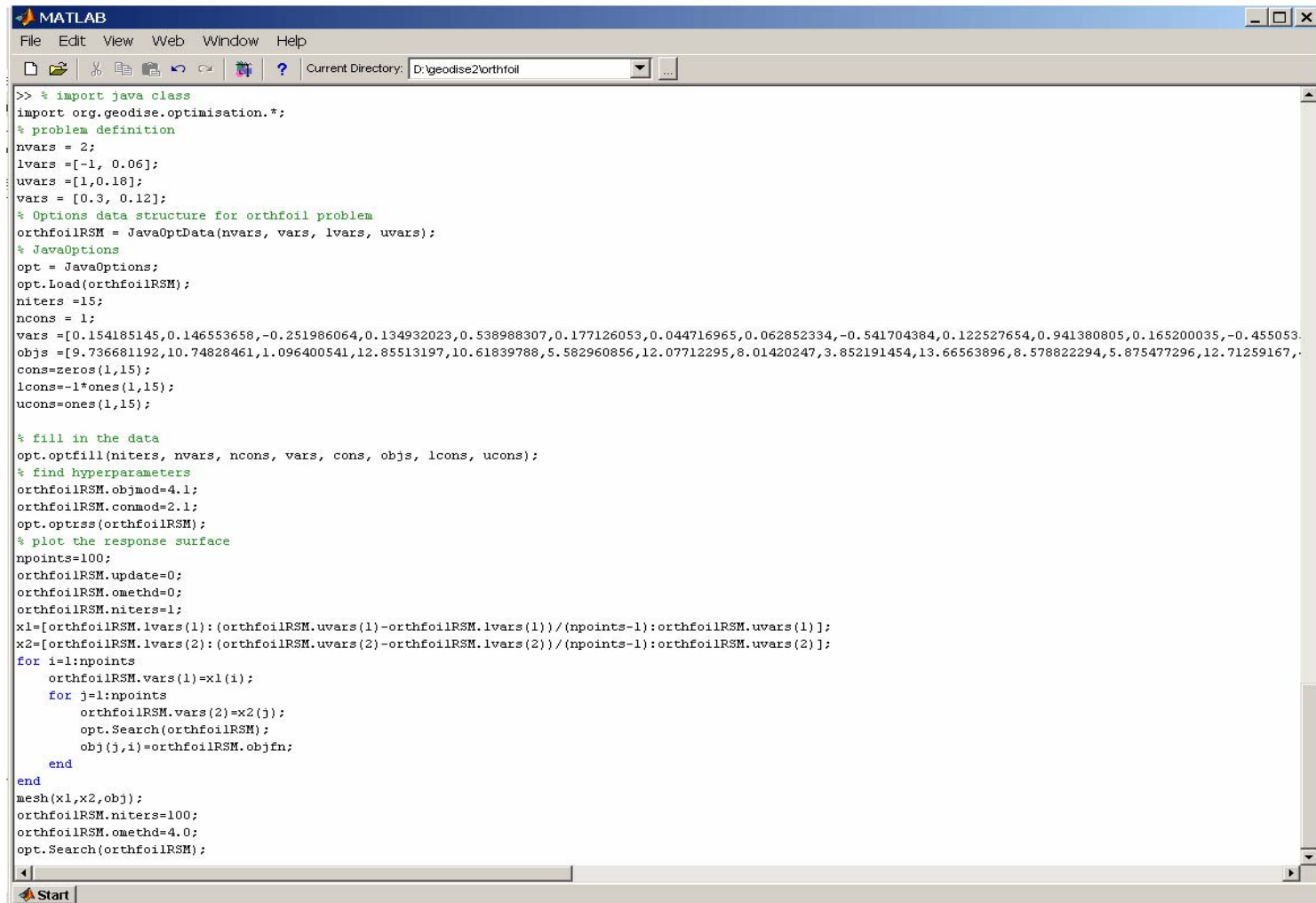


Grid-Enabled Toolkits in Matlab

	Function Name	Descriptions
Proxy management {	gd_createproxy	Creates a Globus proxy certificate from the user's credentials
	gd_proxyinfo	Returns information about the user's proxy certificate
	gd_proxyquery	Queries whether a valid proxy certificate exists
	gd_certinfo	Returns information about the user's certificate
	gd_destroyproxy	Destroys the local copy of the user's Globus proxy certificate
Job submission {	gd_jobkill	Terminates the GRAM job specified by a job handle
	gd_jobstatus	Returns the status of the GRAM job specified a job handle
	gd_jobpoll	Queries the status of a Globus GRAM job until complete
	gd_jobsubmit	Submits a GRAM job to a Globus server
Data archive {	gd_getfile	Retrieves a file from a remote host using GridFTP
	gd_putfile	Transfers a file to a remote host using GridFTP
	gd_archive	Stores a file in repository with associated metadata
	gd_query	Retrieves metadata about a file based on certain criteria
	gd_retrieve	Retrieves a file from the repository to the local machine
	gd_sendtext	Sends a SMS text message to the specified mobile phone number

Pound, G.E., Eres, M.H., Wason, J.L., Jiao, Z., Cox, S.J., and Keane, A. J., "A Grid –enabled Problem Solving Environment (PSE) for Design Optimisation within Matlab", 17th International Parallel and Distributed Processing Symposium (IPDPS 2003) 22-26 April 2003, Nice, France, 2003

Scripting the optimisation workflow within Matlab



```
MATLAB
File Edit View Web Window Help
Current Directory: D:\geodise2\orthfoil

>> % import java class
import org.geodise.optimisation.*;
% problem definition
nvars = 2;
lvars = [-1, 0.06];
uvars = [1, 0.18];
vars = [0.3, 0.12];
% Options data structure for orthfoil problem
orthfoilRSM = JavaOptData(nvars, vars, lvars, uvars);
% JavaOptions
opt = JavaOptions;
opt.Load(orthfoilRSM);
niters = 15;
ncons = 1;
vars = [0.154185145, 0.146553658, -0.251986064, 0.134932023, 0.538988307, 0.177126053, 0.044716965, 0.062852334, -0.541704384, 0.122527654, 0.941380805, 0.165200035, -0.455053,
objs = [9.736681192, 10.74828461, 1.096400541, 12.85513197, 10.61839788, 5.582960856, 12.07712295, 8.01420247, 3.852191454, 13.66563896, 8.578822294, 5.875477296, 12.71259167,
cons=zeros(1,15);
lcons=-1*ones(1,15);
ucons=ones(1,15);

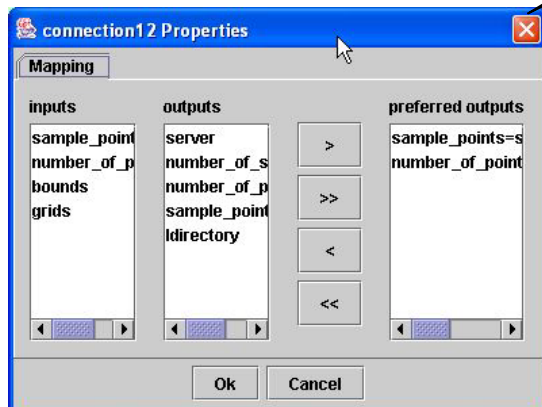
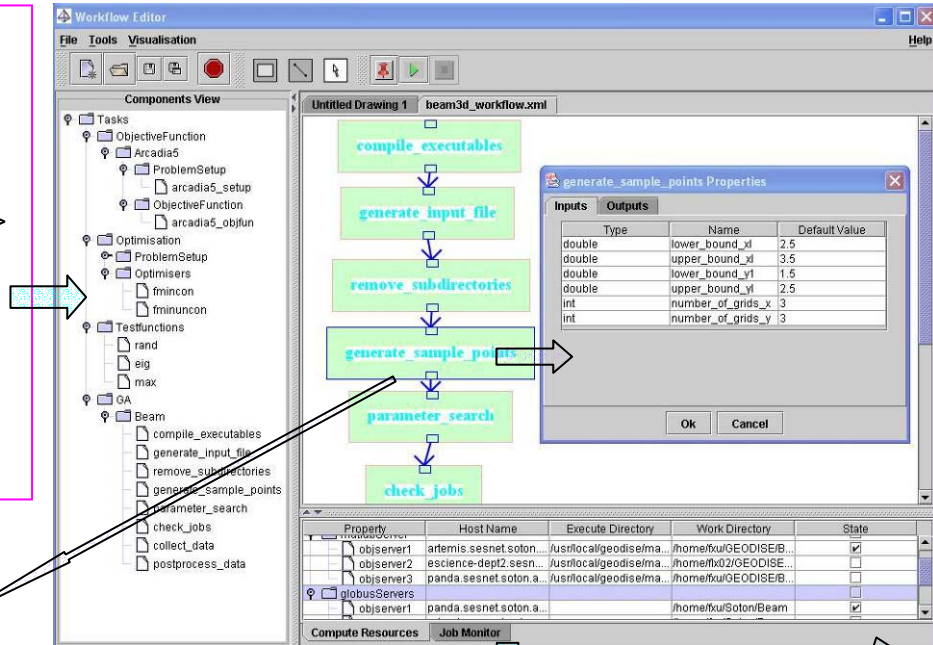
% fill in the data
opt.optfill(niters, nvars, ncons, vars, cons, objs, lcons, ucons);
% find hyperparameters
orthfoilRSM.objmod=4.1;
orthfoilRSM.conmod=2.1;
opt.optrss(orthfoilRSM);
% plot the response surface
npoints=100;
orthfoilRSM.update=0;
orthfoilRSM.omethd=0;
orthfoilRSM.niters=1;
x1=[orthfoilRSM.lvars(1):(orthfoilRSM.uvars(1)-orthfoilRSM.lvars(1))/(npoints-1):orthfoilRSM.uvars(1)];
x2=[orthfoilRSM.lvars(2):(orthfoilRSM.uvars(2)-orthfoilRSM.lvars(2))/(npoints-1):orthfoilRSM.uvars(2)];
for i=1:npoints
    orthfoilRSM.vars(1)=x1(i);
    for j=1:npoints
        orthfoilRSM.vars(2)=x2(j);
        opt.Search(orthfoilRSM);
        obj(j,i)=orthfoilRSM.objfn;
    end
end
mesh(x1,x2,obj);
orthfoilRSM.niters=100;
orthfoilRSM.omethd=4.0;
opt.Search(orthfoilRSM);

Start
```

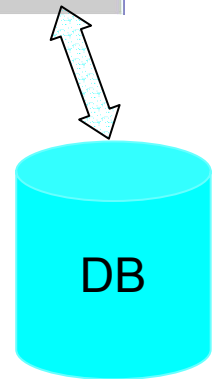
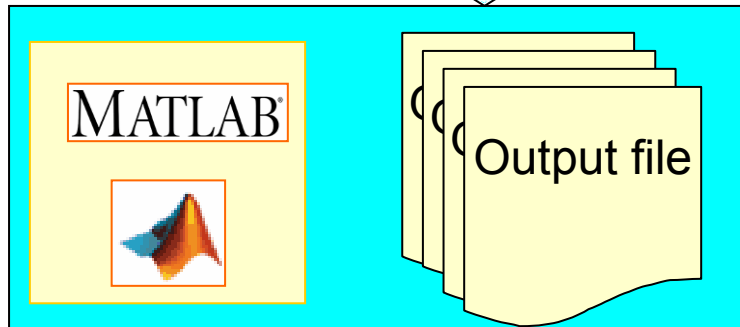
Workflow

Workflow Editor

```
<?xml version="1.0" encoding="UTF-8"?>
...
<Functions>
  <Testfunctions>
    <rand type="Instance">
      <inputs>
        <input1 type="int" name="n" value=" 5 "/>
      </inputs>
      <outputs>
        <output type="matrix" name="A"
value=""/>
      </outputs>
    </rand>
  </Testfunctions>
</Functions>
```



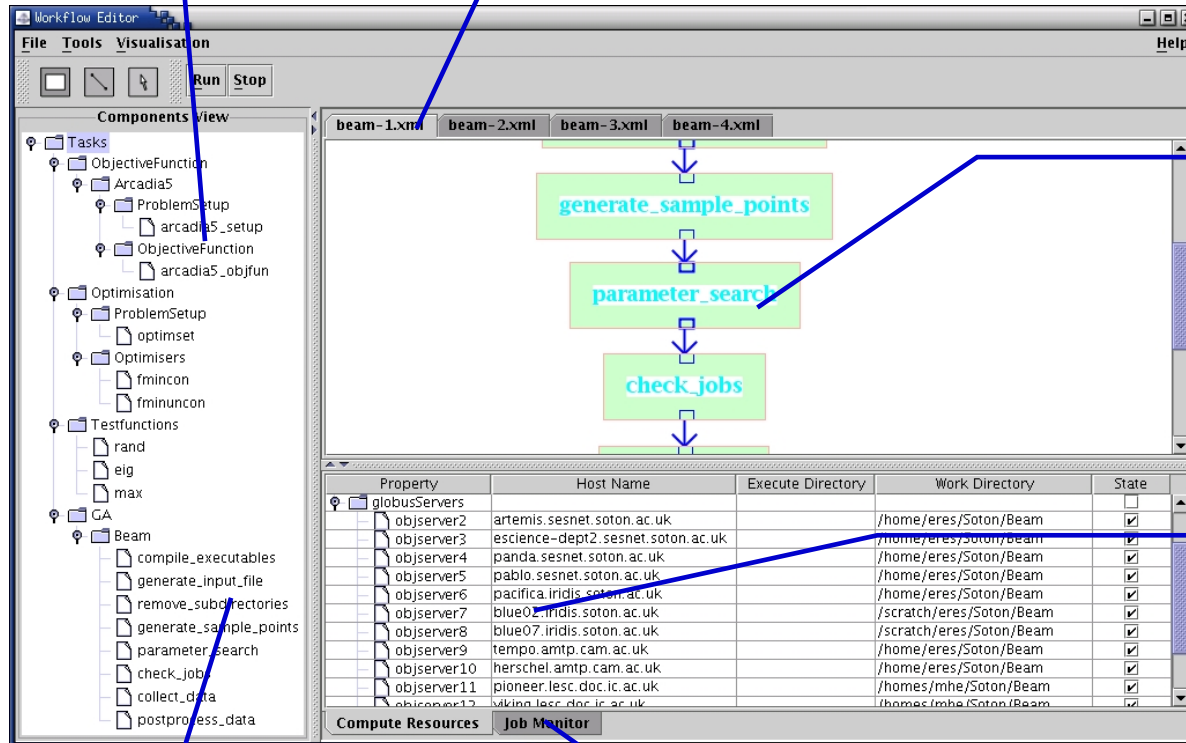
Submit to Matlab



Workflow Tool (Part i)

Retrieve

Projects



Composition Area

Execution - Mapping to Resources

Functions

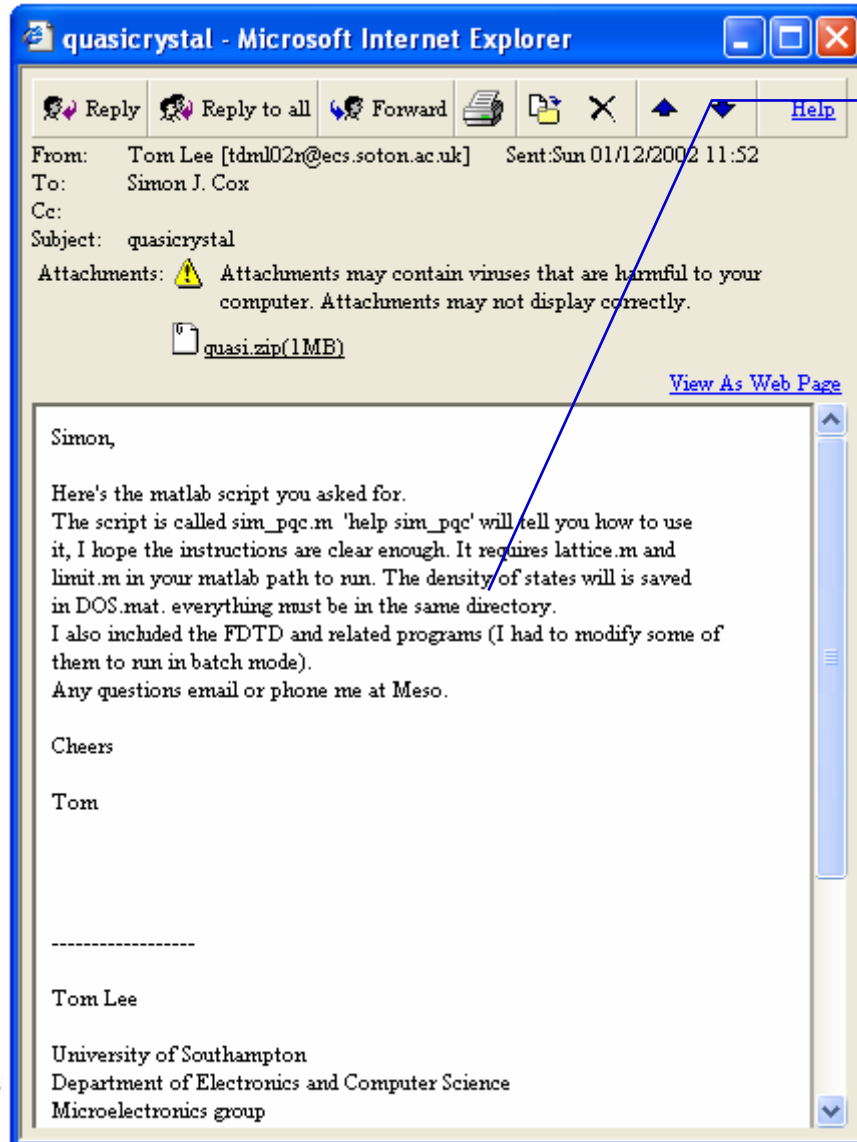
Monitoring and Steering



A few of my favourite things to do with workflows- review

- ✓ Create
- ✓ Retrieve
- ✓ Cut 'n' Shut
- ✓ Configure
- ✓ Execute
- ✓ Monitor
- Share
- Steer
- Dynamically modify

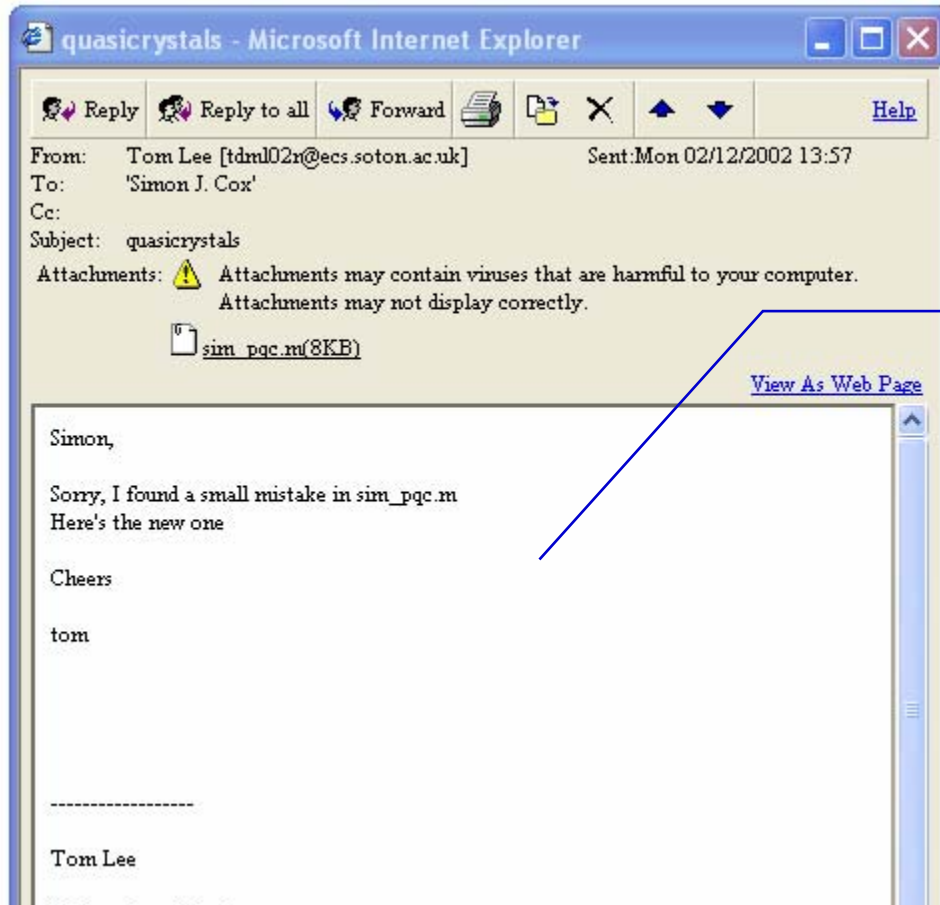
Sharing (i)



Here's the matlab script you asked for.
The script is called `sim_pqc.m` 'help `sim_pqc`' will tell you how to use it, I hope the instructions are clear enough. It requires `lattice.m` and `limit.m` in your matlab path to run. The density of states will be saved in `DOS.mat`. everything must be in the same directory.
I also included the FDTD and related programs (I had to modify some of them to run in batch mode).
Any questions email or phone me at Meso.



Sharing (ii)



Simon,

Sorry, I found a small mistake
in sim_pqc.m
Here's the new one

Cheers

tom

Semantics & Knowledge

Semantic Grid in e-Science

- Bridging the gap:
 - Grid: seamless access to distributed computation and data resources
 - e-Science: distributed collaboration and reuse of knowledge and resources
 - Semantic Grid: Semantic Web technology applied on Grid application
- Building ontology and semantically enrich resource for reuse and management
 - Resource is semantically meaningful
 - Expressed using a standard conceptualization
 - Which is well recognized within a specific community of practice.

Knowledge Technologies

Six challenges define the Life Cycle:

Acquire • Model • Reuse • Retrieve • Publish • Maintain knowledge



Knowledge Acquisition (KA)

Knowledge sources

Domain experts, software manuals & textbooks.

KA techniques

Interview, protocol analysis, concept sorting etc.

Tools used

PC-PACK integrated knowledge engineering toolkit

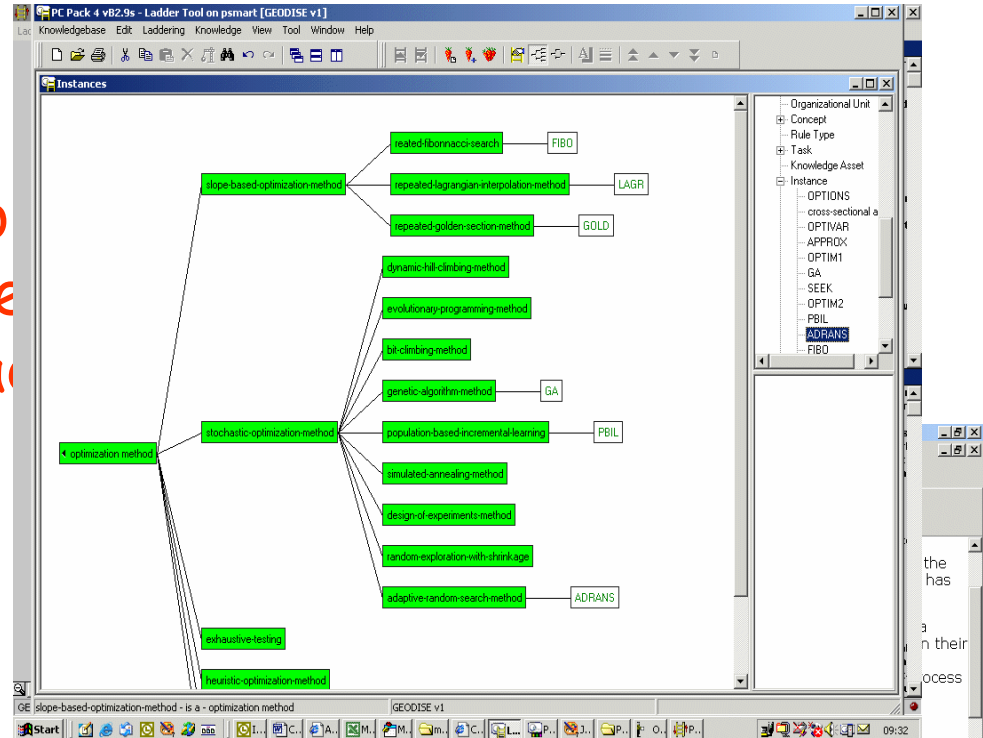
Knowledge acquired

EDSO domain knowledge, EDSO processes and problem definition



Co
hie
La

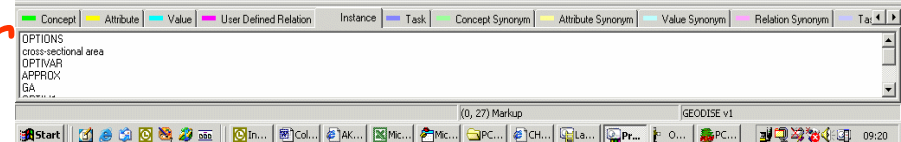
Concep
up in Pr
Editor



5.2. Choice of Optimizer

Here, as has already been noted, the initial input data for the design do not allow all the criteria used as constraints to be met (i.e., it is not a feasible design). This restricts the choice of optimizer since some do not work when given an initially infeasible design. However, limits have also already been chosen for the design variables and the optimizers will therefore seek to remain within these limits, reducing the scope of the task somewhat (some of the methods within the OPTIVAR library treat these range constraints as advisory by default and may, therefore, explore beyond their limits; this may be over-ridden if desired). To give a representative cross-section of the methods available the following techniques have been applied:

- (1) the method of successive linear approximation -APPROX;
- (2) the genetic algorithm -GA, with a one pass, external penalty function -OPTIM1 and
- (3) the Hooke and Jeeves direct search -SEEK, with the Blacoo-McCormick combined external and internal penalty function -OPTIM2



Knowledge Modelling

Techniques

CommonKADS knowledge engineering methodologies.

Knowledge models

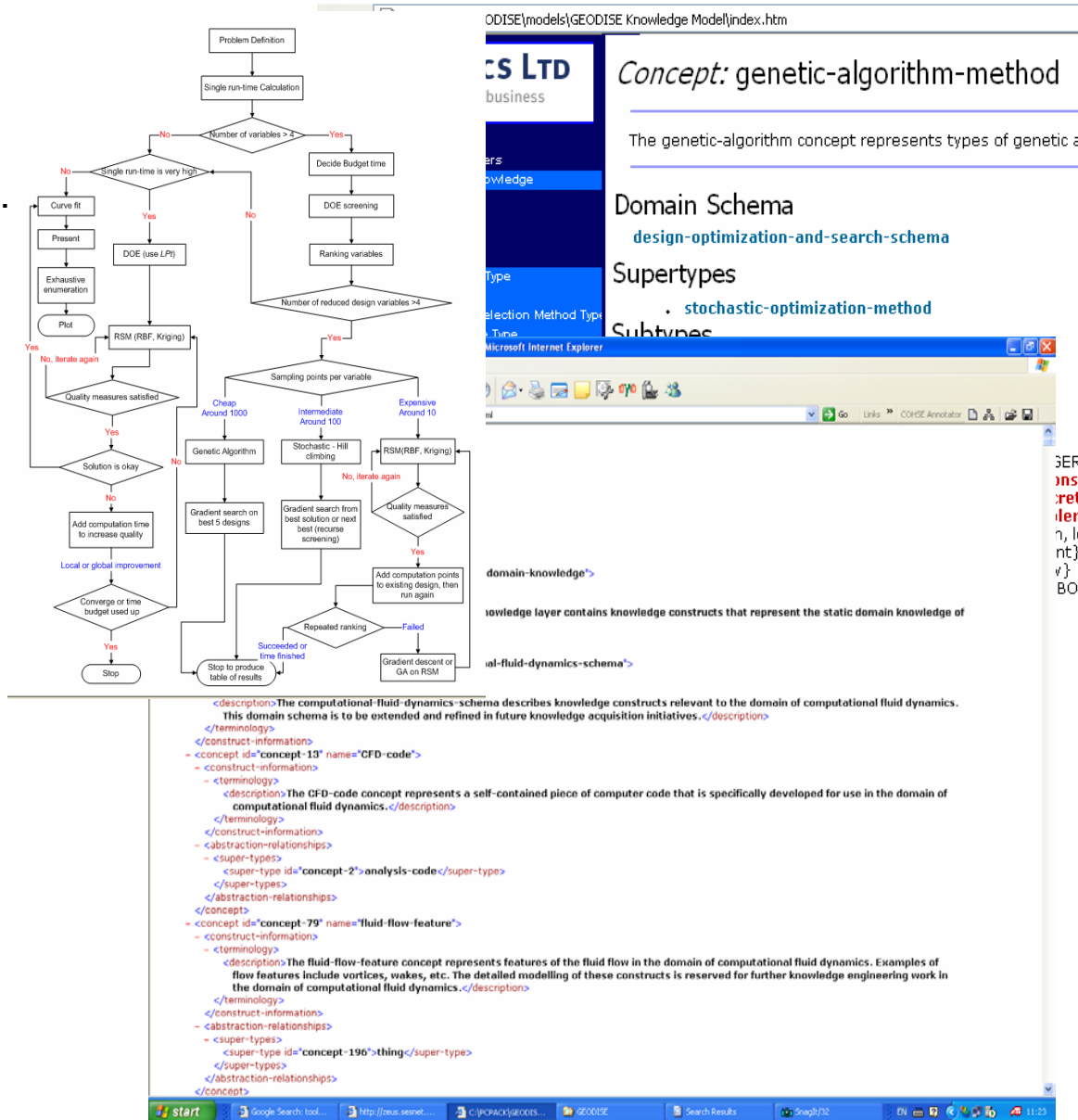
Organization, agent & task templates, domain schema inference rules.

Tools used

PC-PACK integrated knowledge engineering toolkit

Deliverables

Knowledge web in HTML, XML and UML, Conceptual task model, EDSO process flowchart



Ontologies

- common conceptualisation of a domain -



Semantic Workflow support in Geodise

- **Ontology modelling**
 - Definition: Domain conceptualisation that collect a controlled set of vocabulary and their relationship through hierarchy and explicitly expressed properties.
 - Examples: User profile ontology, Problem profile ontology, Task ontology, etc.
- **Instance generation**
 - Definition: Semantic enriching instances by referencing to ontology files
 - Methods: annotation content with ontology, populating ontology with content
- **Semantic consumption**
 - Ontology driven instance querying
 - Ontology driven from generation
 - task configuration
 - Problem setup
 - Ontology assisted domain script editing
 - Service oriented workflow composing – querying semantic enriched service component.



Workflow Tool

(... with added semantics)

Ontology

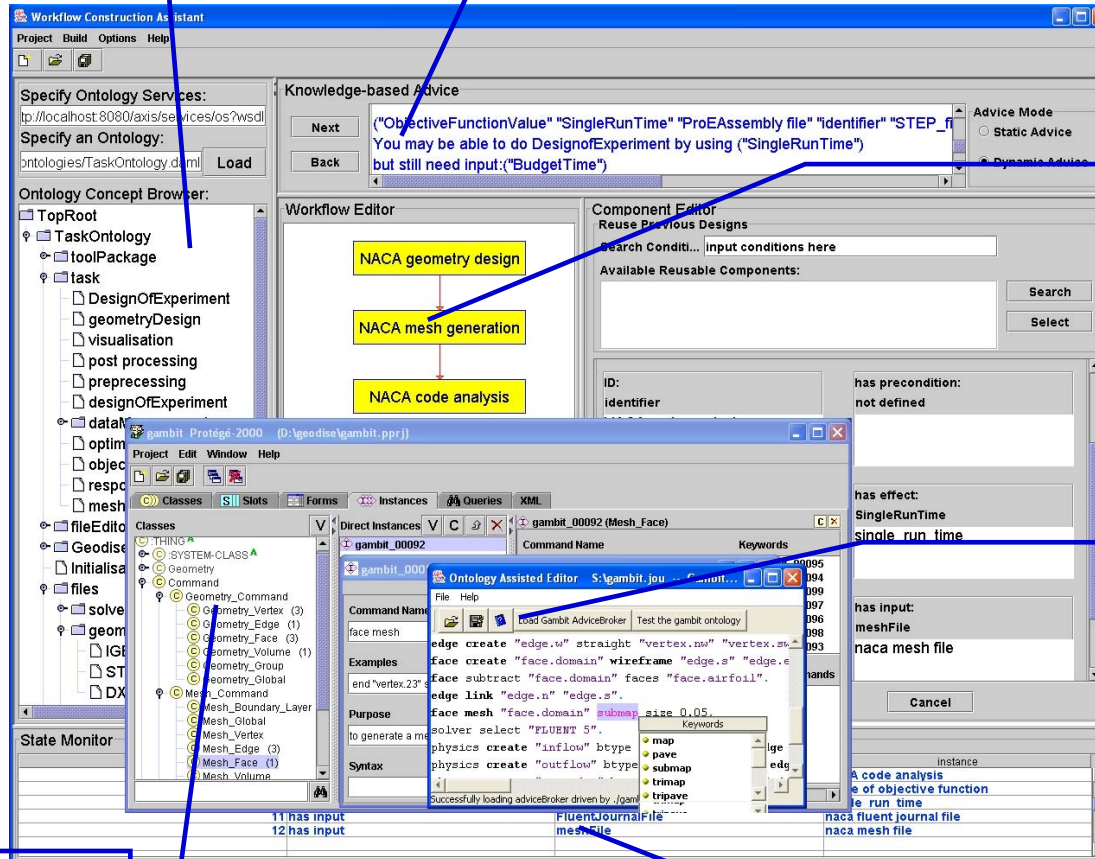
Composition Advice

Composition Area

Advice in Domain Editor

Functions

State Monitor



Ontology Development (1)

- Tools
 - Protégé & OilEd Editor
- Representation
 - DAML+OIL & CLIPS
- Deliverables
 - EDSO domain ontology
 - EDSO task ontology
 - Mesh generation tool (Gambit software) ontology
 - User-profile ontology

Protégé Editor

Oiled Editor

DAML+OIL

```

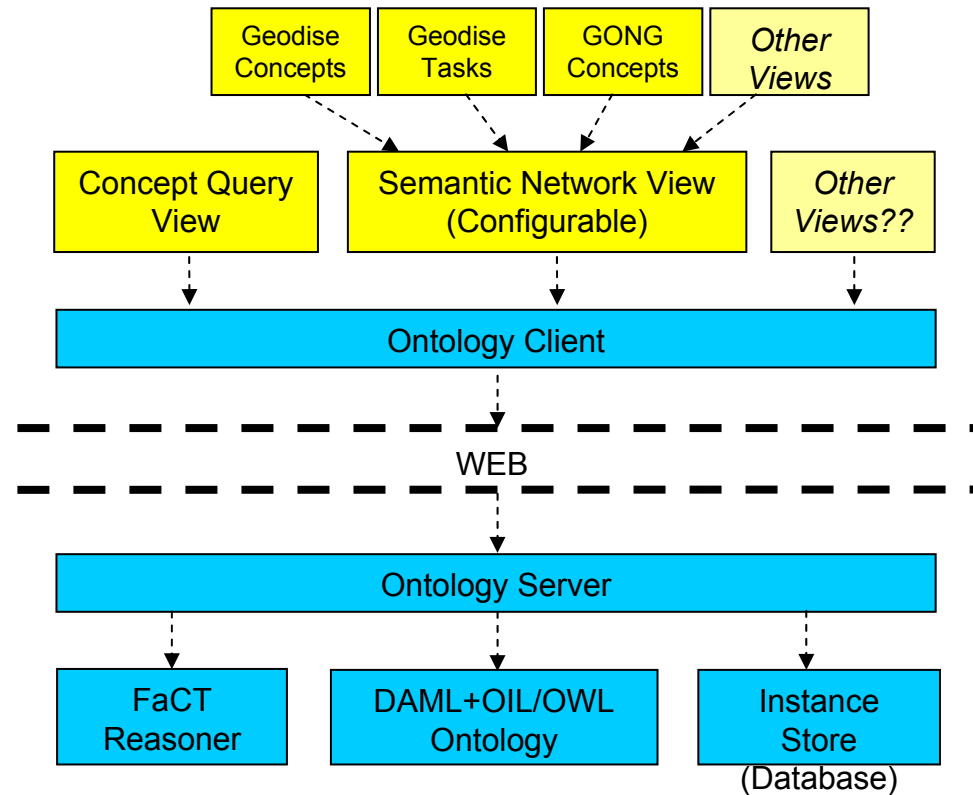
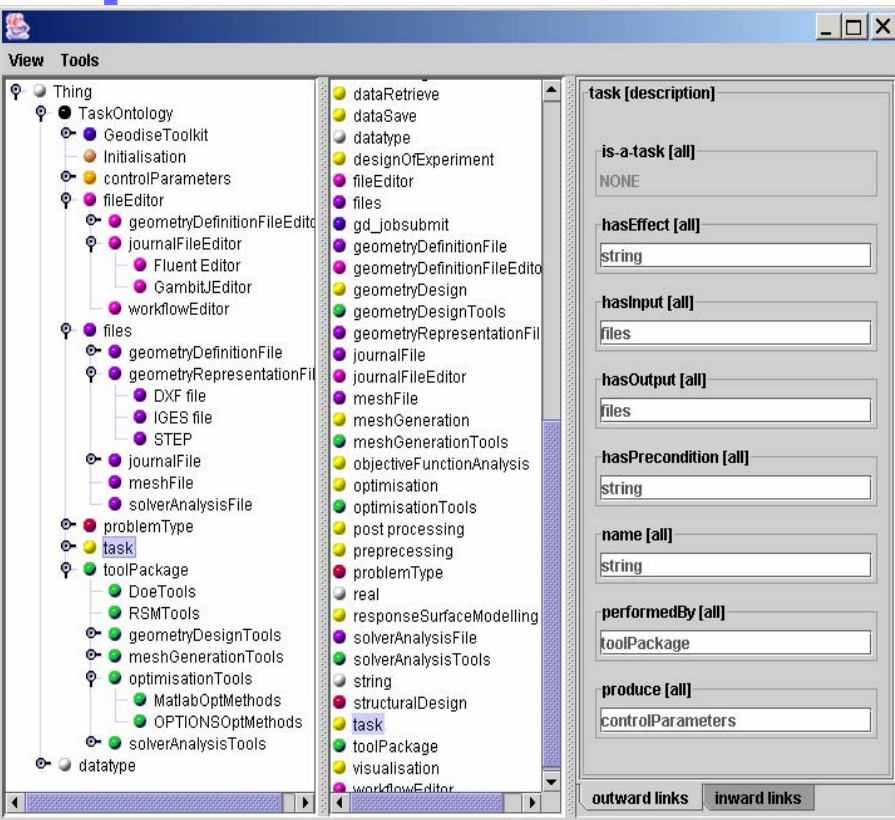
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#geometryDefinitionFile" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#geometryRepresentationFile" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#graph" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#journalFile" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#meshFile" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#solverAnalysisFile" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#AVS_File" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#problemType" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#task" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#DesignOfExperiment" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#dataManagement" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#designOfExperiment" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#meshGeneration" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#objectiveFunctionAnalysis" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#optimisation" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#postProcessing" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#preprocessing" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#responseSurfaceModelling" />
<rdf:type rdf:resource="http://www.geodise.org/KnowledgeManagement/ontologies/TaskOntology.daml#singleRunTime" />

```

type	property	filler
has-class	has input #1	meshFile #1
has-class	has input #1	journalFile #1
has-class	has output #1	solverAnalysisFile #1
has-class	performed by #1	solverAnalysisTools #1
has-class	has effect #1	ObjectiveFunctionValue #1
has-class	has effect #1	SingleRunTime #1

Ontology Development (2)

- **Ontology Views**
 - DL ontologies (DAML/OWL)
 - Simplified views
 - Tailored to specific domains



- **Ontology Views**
 - Underlying complexity hidden
- **Ontology editing by...**
 - Knowledge engineers
 - Domain experts

Ontology Services

- Facilitating ontology sharing & reuse
 - Ontology service APIs
- Domain independence
 - DAML+OIL/OWL standards
- Soap-based web services -WSDL
- Java, Apache Tomcat & Axis technologies



Ontology Services

This page gives information about the ontology services (OS), including the purpose of the OS, interface specification, where it is and how to use it in your applications.

The purpose of Ontology Services

The ontology services provide Java API tools for common ontological operations, such as subsumption checking, retrieving definitional information, navigating concept hierarchies, and retrieving lexical information. By adopting the emerging web ontology standard, DAML+OIL as its underlying representation language, the services can be used with anybody else's ontologies and metadata repositories as long as they're accessible via URLs. As a result of this, the ontology services become perhaps more of an ontology gateway.

Ontology Service API Specification

Ontology service APIs include CGI interface and SOAP interface. they are described in detail at [OS API Specification](#).

Where it is

Ontology services are housed temporarily in the following server:

CGI interface and tester page:
<http://zeus.sesnet.soton.ac.uk:8080/>

SOAP interface:
<http://zeus.sesnet.soton.ac.uk:8080/axis/services/os?wsdl>

How to use it

SOAP-based ontology services can be used in the same way as any web service by creating proxy classes and then calling its functions. for further information about web service, please refer to relevant websites.

Ontology Services - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address <http://portal1/Geodise2/KnowledgeServices/OntologyServices/os.htm>

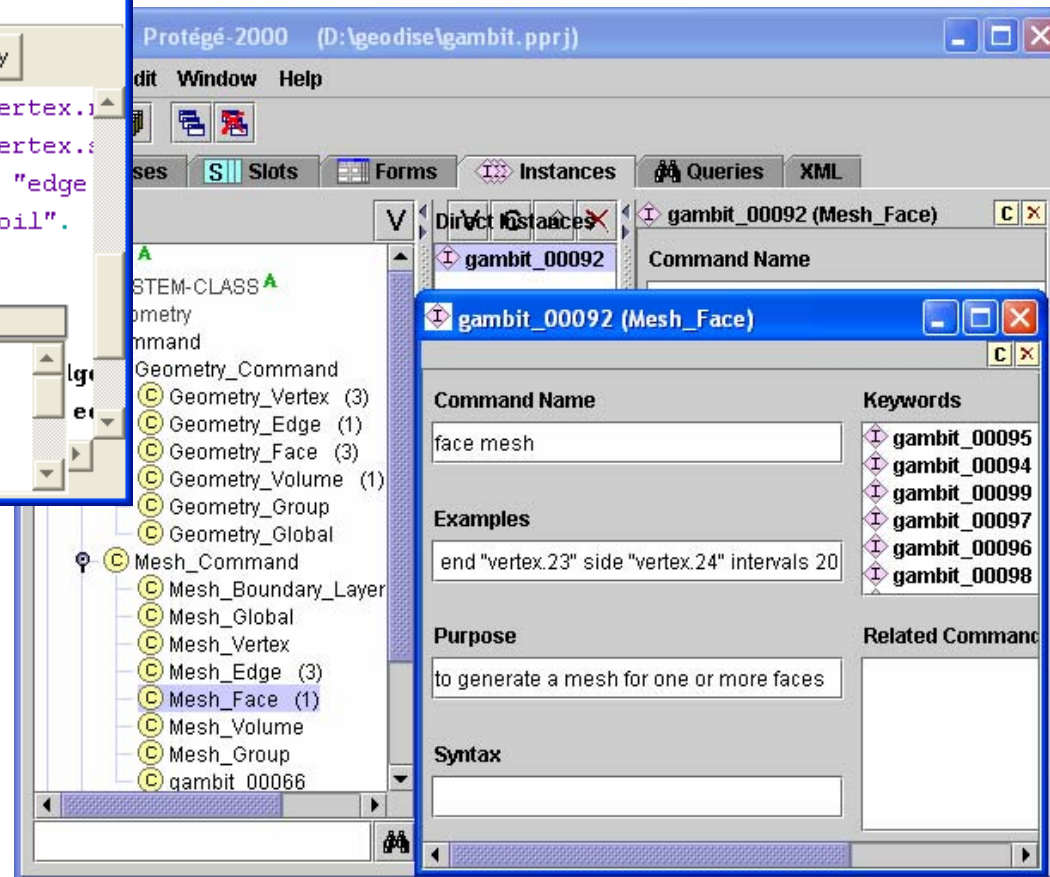
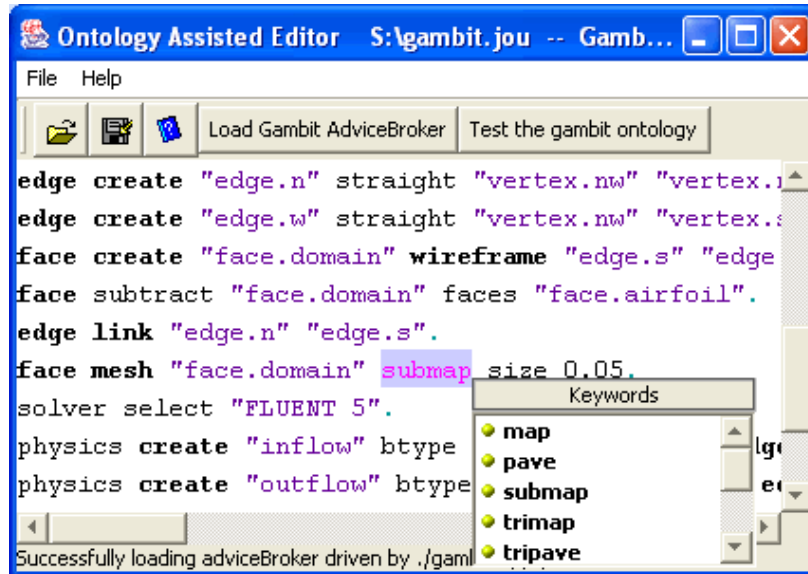
http://zeus:8080/os/OntologyService

File Edit View Favorites Tools Help

Address <http://zeus:8080/os/OntologyService>

```
<?xml version="1.0" ?>
<RESULT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#Time_slot_constraint</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#Design_variable</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#Polynomial_Response_Surface</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#SEEK_OPTVAR</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#davidon_fletcher_powell_method</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#POWELL_NUM-Rcp</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#davidon-fletcher-powell-method</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#One_pass_external_function-OPTIM1</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#CertIssuer</CONCEPT>
  <CONCEPT ontology="http://zeus.sesnet.soton.ac.uk:8080/ontologies/edso.daml">http://protege.stanford.edu/whole#Exhaustive enumeration</CONCEPT>
```


Ontology Assisted Domain Script Editor



- Pre-defined command syntax ontology with Gambit command syntax instances
- Semantic rich instances being consumed in the editor
- Syntax colorizing and auto-completion

Ontology Driven Forms in Geodise -1

- Setting up problems (a scenario using JaxFront)

http://80.254.166.28/jaxfront-v121/servlet/demoservlet?frameHandler=mainFrameSet&sessionID=4341161 - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://80.254.166.28/jaxfront-v121/servlet/demoservlet?frameHandler=mainFrameSet&sessionID=4341161

ProblemProfile

designVariables(s)

designVariables

designVariables

description Beam design problem

timeCreated 2001-12-17T09:30:47

lastTimeUsed 2003-02-17T09:30:47

user barry

dg_id

parameters

	name	unit	meaning	value
<input type="checkbox"/>	length	mm	length of the beam	250
<input type="checkbox"/>	force	N	force on the tip of th	5000.0
<input type="checkbox"/>	density	kg/m3	density of the beam	7850.0

dataFileURL

objectiveFunction

	selected	objective	codeURL	singleRunTime	estimatedLowerBound	estimated
<input type="checkbox"/>	<input type="checkbox"/>	minimize the cross-s	d:\geodise\beam1.s	PT30S	20	30

singlePeak ☒

optimisation not defined

Close Session Show XML Cancel Validate Save

http://80.254.166.28/jaxfront-v121/servlet/demoservlet?xpr - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://80.254.166.28/jaxfront-v121/servlet/demoservlet?xpr

	1	2	3
name	width	breath	material type
meaning	beam width	breath of the beam	beam material
unit	mm	mm	null
fixed	false	false	true

name material_type

meaning beam material

unit null

limit discreteLimit

values

1

2

3

change

fixed ☒

Ontology Driven Forms in Geodise -2

- Configuring tasks in Workflow Composing Environment

The screenshot displays the 'Workflow Construction Assistant' window, which is used for configuring tasks in the Geodise environment. The interface is divided into several panes:

- Specify Ontology Services:** Shows the URL 'tp://localhost:8080/axis/services/os?wsdl' and a 'Load' button.
- Specify an Ontology:** Shows the path 'bntologies/TaskOntology.daml' and a 'Load' button.
- Ontology Concept Browser:** A tree view showing the ontology structure, including 'TaskOntology', 'toolPackage', 'task', 'DesignOfExperiment', 'geometryDesign', 'visualisation', 'post processing', 'preprocessing', 'designOfExperiment', 'dataManagement', 'optimisation', 'objectiveFunctionAnalysis', 'responseSurfaceModell', 'meshGeneration', 'fileEditor', 'GeodiseToolkit', 'Initialisation', 'files', 'solverAnalysisFile', and 'geometryRepresentation'.
- Knowledge-based Advice:** A text area displaying advice: '("ObjectiveFunctionValue" "SingleRunTime" "ProEAssembly file" "Identifier" "STEP_f... You may be able to do DesignofExperiment by using ("SingleRunTime") but still need input:("BudgetTime")'.
- Workflow Editor:** A diagram showing a workflow with three steps: 'NACA geometry design', 'NACA mesh generation', and 'NACA code analysis', connected by arrows.
- Component Editor:** A form for configuring a component. It includes fields for 'ID' (Identifier: NACA code analysis), 'has precondition' (not defined), 'has effect' (ObjectiveFunctionValue: value of objective function, SingleRunTime: single run time), and 'has input' (journalFile: naca fluent journal file, meshFile: naca mesh file).
- State Monitor:** A table showing the state of the workflow components.

index	ID	property	type	instance
8	ID		identifier	NACA code analysis
9	has effect		ObjectiveFunctionValue	value of objective function
10	has effect		SingleRunTime	single run time
11	has input		FluentJournalFile	naca fluent journal file
12	has input		meshFile	naca mesh file

Exploiting Knowledge in Geodise



Knowledge Application 1: Create Semantic Content

- Goals
 - Machine understandable information
 - Facilitate sharing & reuse

- Techniques & tools

- OntMat-annotizer
- Geodise Ontologies

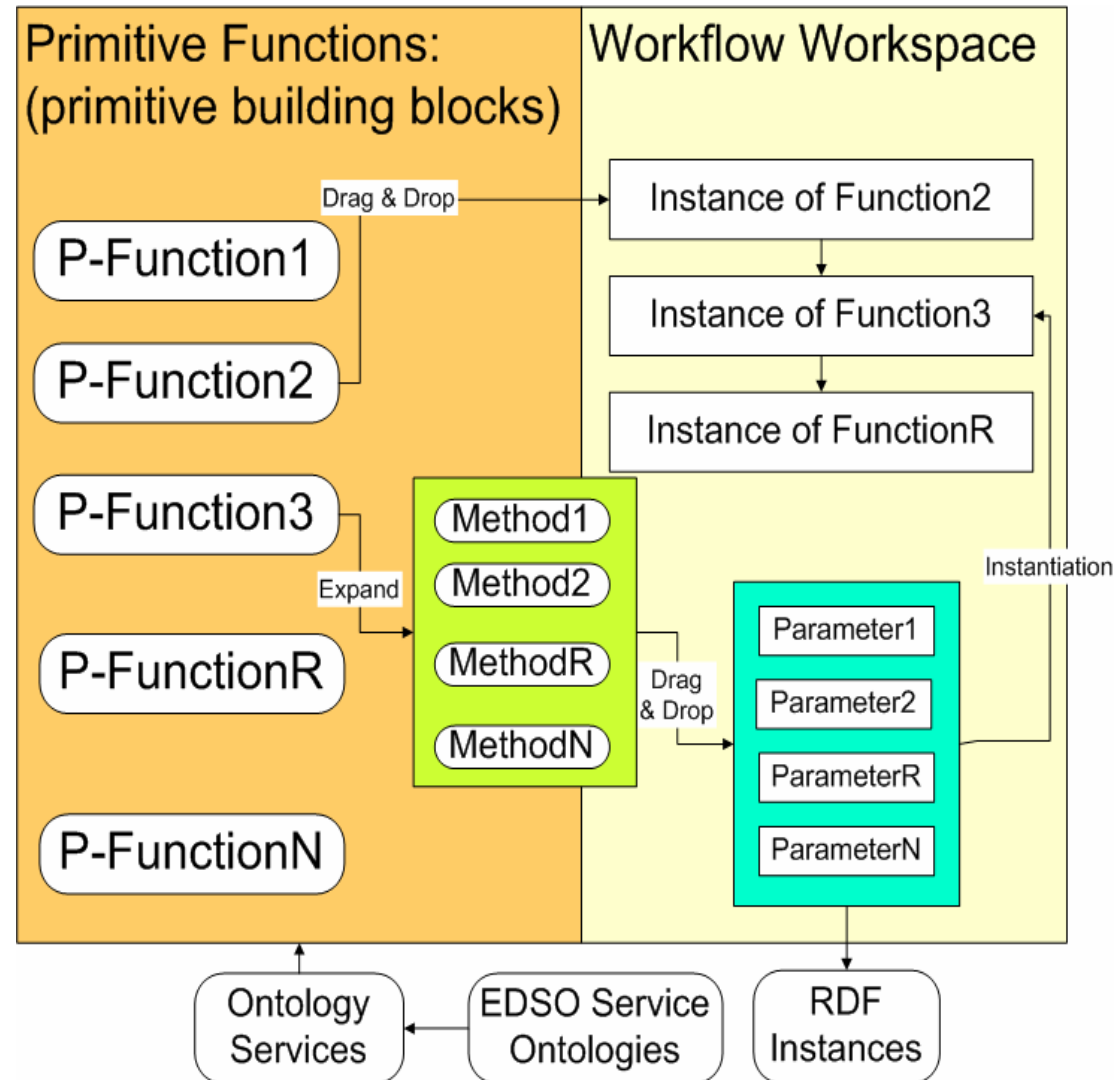
- Example
 - OPTIONS log-files annotation

The screenshot displays a web application interface for semantic content creation. It consists of several windows:

- Ontology Browser:** A tree view on the left showing a hierarchy of concepts. The 'User' node is selected, and its sub-nodes are expanded, showing 'constraint_function (Constraint_F...', 'objective_function (Objective_Fu...', 'problem_designer (User)', 'ajk', 'variable (VariableType)', 'DDP_a_u', 'DDP_xp_j', 'Breadth', 'DDP_a_u', 'DDP_xp_j', and 'Height'.
- HTML Browser 1:** A window showing the 'Design Search' results. It includes a table with 'Attributes' and 'Values' for 'complexity' (simple), 'description' (design of nacelle), and 'problem_ID'. Below the table, there is a 'Design Search' section with a table of results. The 'Design Search' section also includes a 'Design output' section with a table of results.
- annotatedGeodise2wf.html - Notepad:** A window showing the HTML output of the design search. It contains XML markup for the design search results, including the 'Design Search' section and the 'Design output' section.

Ontology-assisted Workflow Management

- **Features:**
 - Function selection
 - Function instantiation
 - Database schema
 - Semantic instances
 - Semantic workflow
- **Technologies:**
 - EDSO ontologies & ontology services
 - Java JAX-RPC, DOM/SAX



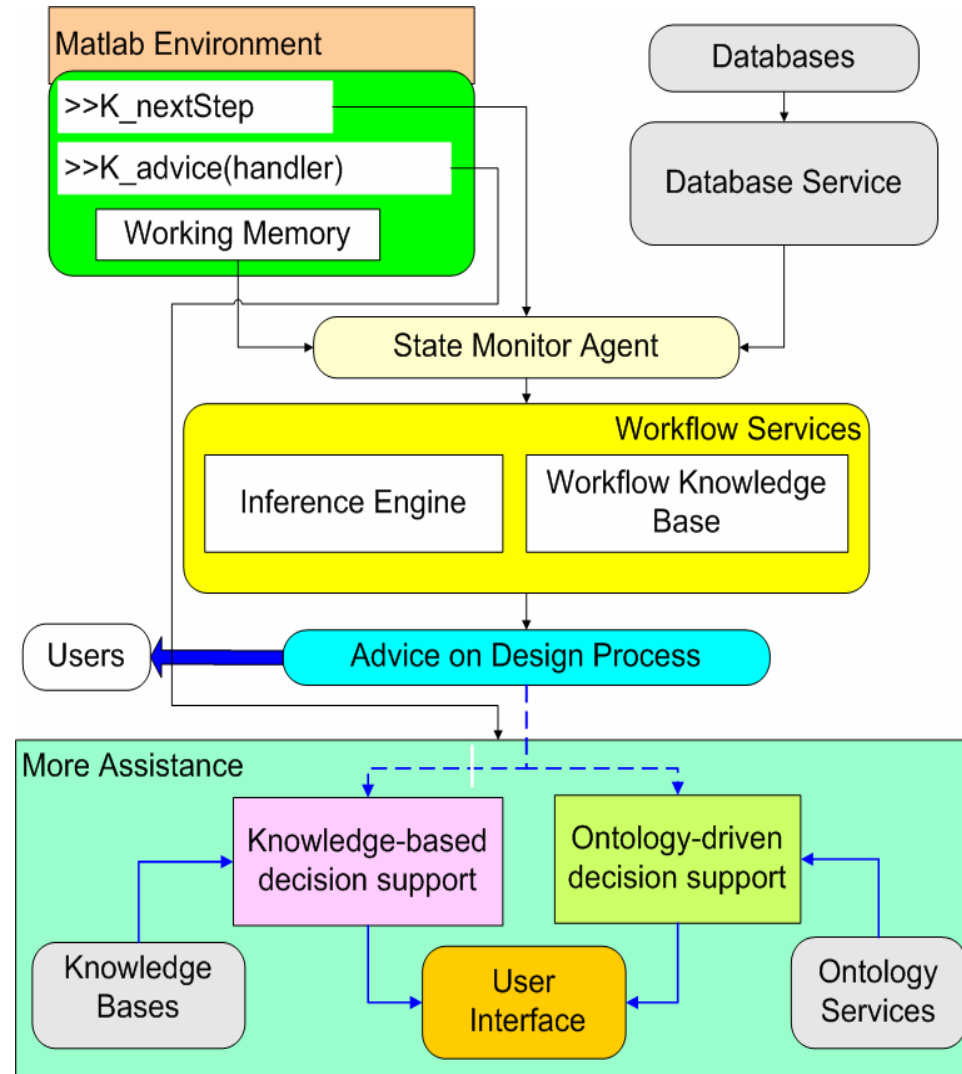
Knowledge-based Design Advisor

- Features

- Context-sensitive advice
- Advice at multi-levels of granularity (process, task ...)
- KBSs as knowledge services

- Technologies

- Knowledge engineering
- EDSO ontologies
- Rule-based reasoning techniques



Intelligent Workflow Monitoring and Advice

(“rule-based to case-based in real-time”)

- Updating constructed workflow using rule-base
 - At run-time:
 - ☐ find ‘similar’ workflows to the one constructed
 - ☐ is this one performing ‘as expected’? Might a different workflow outperform current one?
 - Resolution: Perhaps problem is anomalous?
 - ☐ Change method/ modify workflow?
 - ☐ Feedback to expert ... update rule-base?
- Exploiting new components in workflows
 - Example
 - ☐ New optimisation method added in semantically consistent way
 - ☐ Workflows constructed (by expert) with new method, ... and then:
 - ‘*similar workflow*’ search above will find workflows with new method in:
 - ☐ Might they outperform the currently constructed workflow?
 - ☐ Substitute new method into constructed workflow?
 - ☐ Feedback to expert ... update rule-base?



A few of my favourite things to do with workflows- review (ii)

- ✓ Create
- ✓ Retrieve
- ✓ Cut 'n' Shut
- ✓ Configure
- ✓ Execute
- ✓ Monitor
- ✓ Share
- ✓ Steer
- ✓ Dynamically modify

Questions

Example Script

```
hostname = 'pacifica.iridis.soton.ac.uk'  
jobmanager = [hostname, '/jobmanager-fork']  
rsl = '&(executable="/bin/date")(stdout="remote.txt")'
```

```
%Create a proxy certificate
```

```
gd_createproxy
```

```
%Submitting a globus job and returning handle
```

```
handle = gd_jobsubmit(rsl, jobmanager)
```

```
%Polling the job
```

```
gd_jobpoll(handle)
```

```
%Getting the standard output
```

```
gd_getfile(hostname, 'remote.txt', 'local.txt');
```

```
%Print the output to screen
```

```
type('local.txt')
```