Automake Physics: Random Craft Production

Unver Ertu[1], Atkinson Paul[1] and Marshall Justin[2]

[1]University of Huddersfield, e.unver@hud.ac.uk, p.atkinson@hud.ac.uk
[2]University College Falmouth, justin.marshall@falmouth.ac.uk

ABSTRACT

The project on which this paper is based concerns combining computer based generative systems with craft knowledge and digital production technologies to create new ways of designing and making artefacts which challenge the boundaries between designer, maker and consumer, and between craft and industrial production. This research project involves the development of a digital generative system for the creation of one-off craft/design works based on randomly generated 3D matrices. The system is to be used by the maker/researcher involved in the project to develop new work, and most significantly, by consumers who will then become 'co-creators' of their own craft/design works without any CAD or advanced computer skills. The data produced by this new software will then be used to control a range of rapid prototyping/manufacturing machines, which will physically produce the new works.

1. INTRODUCTION

Generative systems are now closely associated with the computer as it has provided an ideal environment to codify and execute this type of rule based system. They have been used in many fields in an attempt to model and understand existing natural phenomenon, or as a tool to help find solutions to complex problems. Within art, design and architecture generative systems are being increasingly employed in a range of situations [1,2], not only to extend our understanding of natural structures and solve problems, but also to generate completely novel forms with new aesthetic characteristics. The term generative system applies to any system in which a few basic rules are repeatedly employed to produce varied, unpredictable and often complex results. Implicit within the use of these systems is the recognition that the user relinquishes some control over the end results as the system runs autonomously.

The origins of this project began in 2002 with experimentation into the application of computer generated random form to 3D product design. Computer based 3D solid models were created that could randomly mutate within parameter envelopes set by the designer. At any given point the mutation could be halted by any user of the system and a real world product generated via digital manufacture [3]. The second phase of the program has been to explore the differences between the use of these type of generative systems by a craftsperson as opposed to a product designer. The mutation of a set form employed in the first part of the program has been replaced with a system connecting small building units together in a random fashion within meshes that can be selected and adapted by the user [4].

In contrast to the consumer's limited input in the systems created by the product designer (i.e. to stop an animation of a mutating form), the craftsperson was interested in providing more opportunities for end-users/consumers to interact with the build systems. For a range of complex reasons people have emotional bonds with successful craftworks and it was speculated that through an individual being personally engaged in the design of an object that an emotional bond with objects generated through the systems developed in this project could be created. Therefore the focus of this

project was not on developing design tools for other designers or craftspeople to use, but developing mechanisms for a broad constituency of individuals to have a sense of engagement in the creation of a new object.

Three dimensional modelling software can be used to produce one-off artefacts [5]. The aim of the software developed here is not to allow users to unrestrictedly design new objects, but allow the modification of the dimensions and shape of forms which are then randomly filled with a variety of predefined building units to create unique objects. Obviously, such an approach is contentious to designers and perhaps even more so for craft practitioners or designer-makers. If a craftsperson is required at the point of production then the outcome becomes dependent on that particular craftsperson's individual skills and the production capacity they have available. This is in contrast with the traditional product design model where the designers' skills are required only to define an article and its manufacture. Peter Dormer has stated that one of the dominant definitions of craft is that it is "a process over which a person has detailed control."[6] The impact of these technologies, therefore, is potentially far more significant for a craftsperson than a designer, as a designer is at least accustomed to little or no involvement in the production process. It is also more significant for the status of the artefacts produced from a consumer's perspective, as they too are accustomed to mass produced products not being associated with the designer, but might expect craft products to be associated with a named craftsperson. In contrast to these challenges to some of the accepted characteristics of craft practices, it could be argued that the digital systems developed within this project propose a new way of creating objects which can be related to the older tradition of bespoke commissioning, but potentially in a more democratic and widely available way. Therefore this type of system has the potential to rekindle and expand a craft tradition in which maker and client work together to develop a design that is unique to the individual.

2. DIGITAL MANUFACTURE

E-manufacturing is concerned with the use of the Internet and e-business technologies in manufacturing industries. It covers all aspects of manufacturing including sales, marketing, customer service, new product development, supplier relationships, logistics, manufacturing, strategy development and so on. Digital production technologies enable the creation of objects directly from computer data. Either by a reductive process of cutting away material using CNC (computer numerically controlled) equipment, or an additive process using RP (rapid prototyping) technologies. These technologies have had a dramatic impact on industrial production and are starting to become more embedded into the practices of independent creative practitioners. They make the physical production of artefacts from on-screen models a realistic proposition and it was this technology that formed the basis for this project [7].

Direct, Digital or Rapid Manufacture (RM) is essentially the adaptation of RP technologies to the manufacture of end-use products. RP technology enables the digital manufacture of prototypes directly from CAD data. Additive manufacturing process uses layer-based "rapid prototyping" technologies for production manufacturing. Additive manufacturing stands in contrast to typical current manufacturing processes, where material is removed or formed. This manufacturing method is not as yet a replacement for mass production, since the economics of RP technologies do not enable cost-effective high volume production. However in the production of one-off works, which in the past has been a significant factor in making craft production distinct from industrial production, RM technologies have the potential to make a significant contribution to this niche area of production.

3. PROJECT CONCEPT

This project investigates the use of generative design processes in theoretical and practice-based research in art, design and architecture and it explores the potential for mathematical algorithms to provide computer generated inputs for the creation of artworks, three-dimensional forms or architectural propositions. Through the 'Post Industrial Manufacturing Systems' project, the University of Huddersfield has been investigating the use of such processes for over five years focusing on their application within product design [8].

The overall aim of this project is to investigate the potentials of using a generative system to facilitate the design of unique one-off works. The specific objectives of this project were to:
• Create software that allows users to easily control the generation of unique and complex matrix structures.
• Produce a method by which these structures can be outputted in a format appropriate for digital production.
• Investigate a range of digital production technologies for their appropriateness to creating craft objects from the data generated.

3.1 Software Development

The development of software is divided into two distinct sections. The first part of the development process is called the development of **Matrix Build** software and the second phase covered by this paper is the development of **Random Fill** software.

Virtools [9] game authoring software was employed throughout this project to create the systems described in this section. It was selected for a number of reasons, including;

- Software development is based on a more intuitive 'flow chart' building method rather than hard scripting.
- It allows the creation of highly functional user friendly interfaces relatively quickly and easily.
- Completed software can be easily published and made available for use via a web browser.
- Player file is relatively small

3.2 Development of Matrix Build Software

For creating unique forms, different methods can be used:

- parametric objects where the dimensions of the object are driven by computational changing parameters as used in FutureFactories [10]
- A modular system where complexity is created through rules being applied to the repetition of simple units to develop a system for building/growing forms rather than using predefined parametric objects. [3]
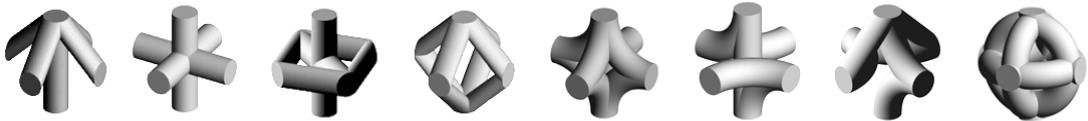

Fig. 1: Building block units.

This developed software gave the user the opportunity to select any of the building block units shown on Fig. 1. When the generative system is started, software selects a unit at random and is placed in one of the free spaces next to the initial unit, the system then checks all the spaces around the units and randomly selects one of the free spaces to place another randomly selected unit. This was achieved by identifying the availability of the adjacent objects by checking collisions. This process continues until the system is stopped and a file saved.
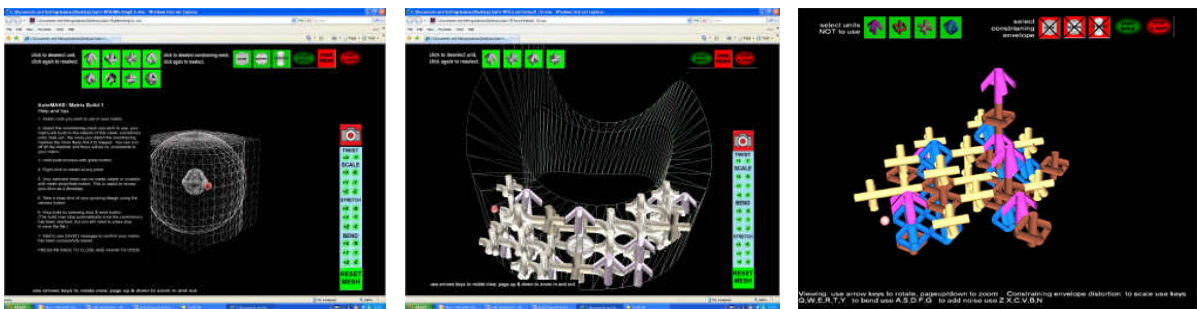

Fig. 2: Matrix build software.

In order to create a greater level of user control and put some restriction on the generation of potentially infinitely large matrices, a series of constraining meshes were introduced. These meshes function by acting as an obstacle to the growth of the matrices. Four different types of meshes were employed: **Cube, Sphere, Cylinder and Torus.** Also the following functions were introduced: **twisting scaling, bending and stretching (**Fig. 2).

As a restrictive mechanism, the constraining meshes have been moderately successful, although if the meshes are heavily distorted then units can often 'leak' beyond the mesh and once this has occurred the matrix will grow unrestricted. Once the matrix has grown to the limits of the constraining mesh (or the user chooses to stop the process),

the form can be reviewed and saved for production. There are a number of significant limitations in this system such as the highly structured rectilinear format of the structures it generates, the unpredictability of the constraining mesh function and the large memory requirements, especially when larger number of units are used (i.e. over 300).

3.3 Development of 'Random Fill' Software

This more recent system was developed in order to counter some of the limitations of the 'Matrix build' software cited above. This system creates structures from the same basic units as the 'Matrix build' software but the mechanism for construction is significantly different. Instead of forms growing through the random placement of units, they are created by dropping units into a hollow form or 'mould'.

3.3.1 Initial Development and Testing

As the next stage of the development process, physics-based simulation is used to create structures from the similar basic predefined units. In this method, forms are created by dropping units into a hollow form, and the position of the units is determined by collisions factors i.e. friction, elasticity, mass, size, and damping factors.

The use of physics engine capabilities within Virtools allowed each group of units to be given a different set of characteristics (e.g. weight, elasticity etc.). In order to limit the memory consumption, initially each unit was represented by a different coloured sphere during the build process (spheres have a much simpler geometry than the actual units to be used in the final forms and therefore less processing power is required to calculate their interactions). Spheres of various scales were dropped into a mould, eventually filling it to produce a structure made up of nested spheres. The use of different scaled spheres helps create both variety in the density of the generated form and greater structural coherency.

The first physics engine-based system utilised a bowl shape mesh. (See Fig. 3).   The spheres, which contain building block units, rotate around an axis to allow random placement of the units. The mesh is placed on a plane where an impulse can be applied by mouse click to shake it, which provides further mixing of the spheres when filling process takes place and aids the close packing of the spheres (in the same ways a s shaking a jar of different sized marbles creates closer packing)
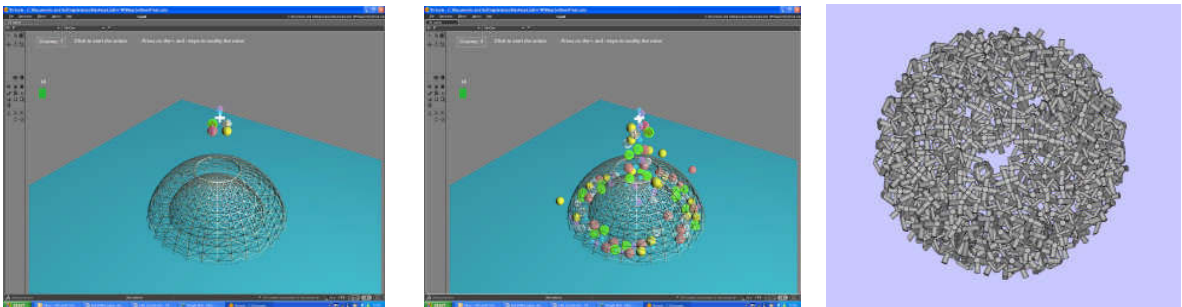

Fig. 3: First 'random' fill software, bowl & render of bowl after post processing.

To provide an example of how complex a form can be created, a pre-existing CAD file of a horse has been employed as the 'mould' and further refinements and optimizations have been undertaken which allow many thousands of units to be used in a single build. After testing the performance, actual building block units are also successfully used rather than spheres with a minimal drop in performance.

There were requirements to test the filling process of more complicated volumes by the building blocks. Initially a female mesh model was placed in a cube in which was filled by the building block units to test the use of more than one surface. The volume between the female body mesh and the internal surfaces of the cube were successfully filled with the units (See Fig. 4). Although this was not a solution to create a designer item it proved that internal and external surfaces can be used at the same time. Then, to test the system, a horse mesh was cut into halves and from two different locations spheres were dropped onto the mesh and successfully filled with the units. A problem occurred

with the position of the legs of the horse, which could not be filled as friction between the surfaces stopped the spheres from moving further than half way as seen in Fig. 5.
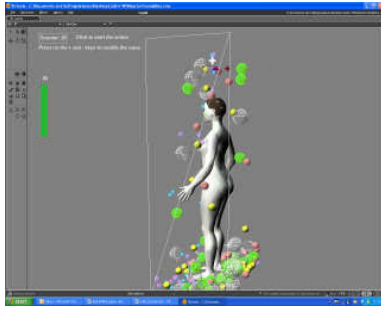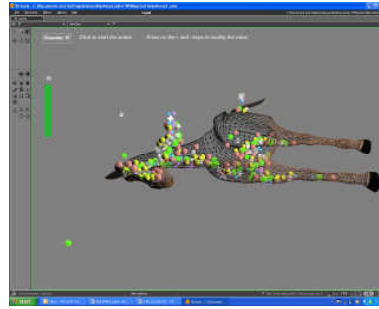

Fig. 4: Female mesh.   Fig. 5: Half horse mesh.

3.3.2 Random Fill System

A library of 3D models was included in the simulation process including a horse, fish, and cat but these could be extended to any supplied cad models. These 3D cad models are generally created in 3d software packages such as 3D Studio max, Poser or Maya, large number of which can be freely downloaded. Fig. 6 shows the developed software which enables user to rotate and zoom the camera view and also hide or reveal the meshes.
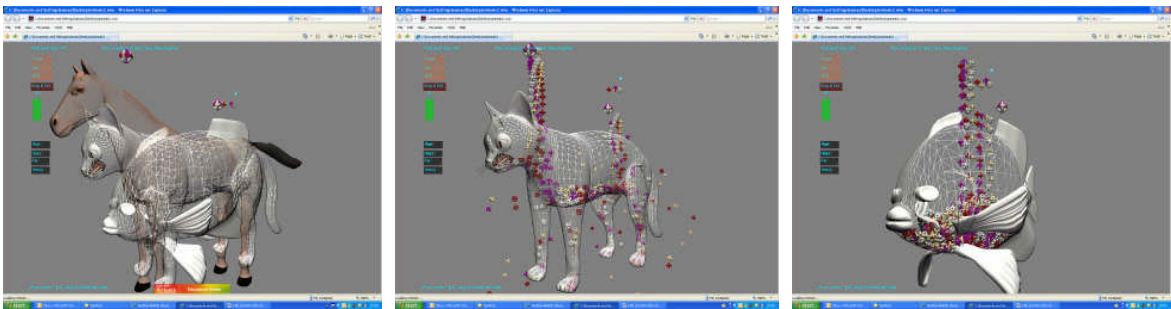

Fig. 6: Horse, cat, fish polygon mesh.

In this paper, the horse model is used to demonstrate the capability of this software and its digital manufacturing process. The horse model created in 3D Studio Max is optimised to have a limited number of polygons which will help to achieve a small file size. Some of the meshes are removed between the head, mane, back and tail of the horse where the building block units enter into the horse body. This 3D model is then imported to a Virtools environment as a Virtools or 3ds file.
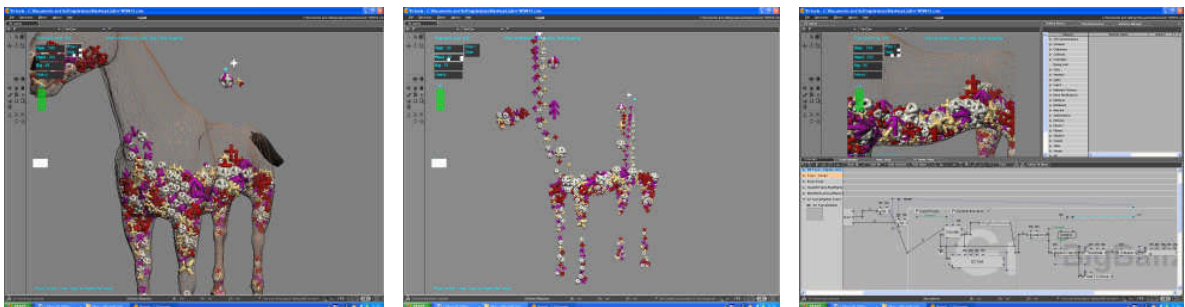

Fig. 7: Horse mesh body and building block units.

Although it is possible to put the building block units through the horse polygon mesh, this method proved to be difficult, especially when the body of the horse is nearly filled, crashing the system. Therefore the holes in the mesh allow the building blocks to overflow and fall when needed. Objects are dropped from predefined locations in a loop. In Fig. 7 the first image shows the horse mesh and building block units and the second image just the units and final image includes a sample of the programming flow chart created in Virtools.

In order to speed up the process the number of units were increased to four from each location and are rotated around an axis to enhance the random fall of units. Larger units were also introduced to give the user greater options and input into the creation of a form. The software interface includes relevant buttons to activate this drop process and to stop the process once 'mould' is filled. On each click of the button 200 units from each location drop into the mesh. Due to elasticity, friction and damping factors of the surfaces, sometimes the leg areas of the horse are not filled by the spheres. So it was recognised that there was a need for moving some of the units. Two methods were used to solve this issue:

- **Nudging:** To increase user interaction within this system, the ability to send an impulse to the horse mesh (i.e. give it a nudge) using the mouse was introduced. This rearranges the units internally without changing the sphere shape. This method has limited usage as most of the building block units are not selectable as the horse mesh body blocked the selection of units.

- **Dropping very heavy objects:** This method uses small blue spheres where their mass parameters were made 20 times heavier than other objects. This mass parameter has now been set as a variable so that as the number of objects increases, they get heavier. Dropping very heavy spheres into the mesh causes all the units close by to be moved because of the impact. Because the blue spheres are heavier, they move until they reach the bottom of the feet of the horse, by moving and pushing the other objects out of the way.

The next process is to output the data from the Virtual to 3D Max environment where repositioning of units needs to be scripted. Initially all the objects were exported as OBJ file format, but due to very large file size issue, a new method was developed. In Virtools **unit names,** x, y, z **positioning** and **angular** coordinates of each unit are written to a text file. These text files are extremely small, even for builds involving many thousands of units, and are therefore easily sent via email. In addition the horse mesh file is also automatically exported in OBJ format. Exporting only the low poly horse mesh, excluding all building block units, eliminated the large file size issue. This system has proved to be very successful, however it did require the creation of a max script to be run in 3D Studio Max that recreates the matrices generated in our build software. 3D Max can then export the structure in a file format appropriate for digital production (i.e. STL).
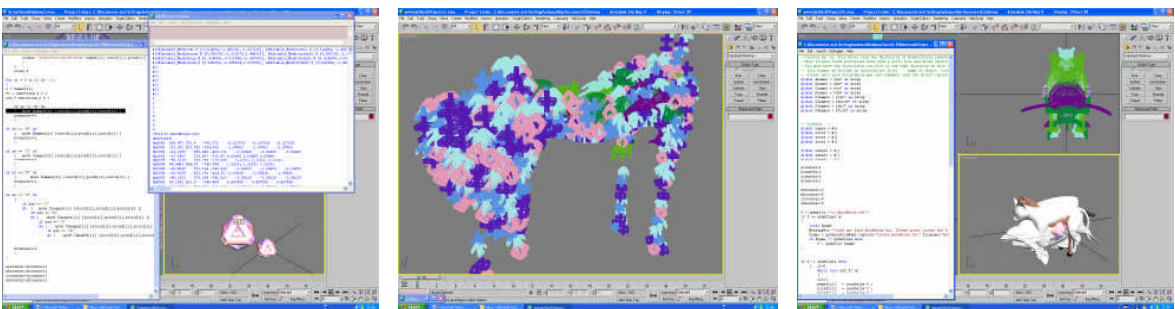

Fig. 8: Creation of shape using script in 3D Max.

Fig. 8 shows the process of rebuilding the horse matrix using the text data. The first image shows the 3D Max script generated and run to build the object from initial building block units. The second image shows the 3D horse model generated using these units by running the script. And the third image shows the trial of using a different type of mesh as building block units such as cat, fish, dolphin and dog.

The production of rapid prototyped or rapid manufactured objects involves using specialist file preparation software. STL models generated by 3D Max of these complex forms, such as the matrices, are rarely perfectly constructed (overlapping surfaces, triangles, holes) and require mending before they can be physically produced. Magics software by Materialise was used for this purpose.

In comparison with the 'Matrix build' system the arrangement of units is considerably more complex and does not create structures which fall into any standard structural format. This system gives a considerably more efficient use of RAM than the 'Matrix build' system as all the individual units' coordinates are not held in the memory at all times. Therefore the number of units that can be used within a single build can be increased by a factor of between 5 before memory usage becomes an issue. All the software systems developed run on a web browser using the freely available '3D life' player produced by Virtools.

3.3.3 RP Objects Manufactured
A range of digital production technologies have been employed within this project. A ZCorp 3D printer, Invision from 3D system and a 3D systems SLS machine have been used to produce some test forms shown in Fig. 9.

Fig. 9: Randomly generated physical objects.

These forms represent three of an infinite range of unique structures which can be designed and built using the 'Matrix build' systems. The test of the appeal of this object generation process will be in the response of users when the systems are made available via the project website. However the intention of the designer was to create a balance between the random generative element and user control within the software. It is hoped that this will give both a sense of engagement in the design process and provide the potential for the unexpected.

4. CONCLUSIONS AND FURTHER DEVELOPMENTS
A range of software has been developed to create unique objects using the physics engine capabilities of Virtools as a mechanism to compile forms in new ways. User testing of these systems is planned to further develop both the user interface and functionality. A web site has been created where software can be tested for the project (www.automake.co.uk) and exhibitions are scheduled to disseminate the work. Further developments planned include:

- Increasing the efficiency of the software's memory use and therefore allowing builds that involve much greater numbers of units.
- Using the physics engine capabilities of Virtools as a mechanism for creating new algorithmic methods that have the potential to compile and reveal embedded orders. These methods could potentially be used to construct forms which are not only aesthetically novel, but also have useful physical characteristics (e.g. structural coherency, increased strength, superior rigidity etc). This project has been undertaken from the perspective of a designer-maker with no background in structural engineering or material science. This avenue of further research will require other collaborative partners with appropriate skills and experience.
- Increasing the choice of build units available to the users by providing a library of 3D CAD models or creating the ability to drop their own models into the design system.
- Merge the Matrix Build and Random Build software to work together.
- The majority of the physical models were produced with RP technologies which were directly available to the research team. Further work on employing new forms of RM technologies, especially those which allow the direct production of objects from metal and ceramic need to be undertaken.

Finally, it should be recognised that the most significant outcome of this project is not the development of the generative systems, it is acknowledged they are not in themselves a significant contribution to the field, but the integration of a number of processes and procedures to create systems that have the potential to engage individuals in a form of digital design and production that questions their familiar relationship with consumer products.

As and when RM and desktop manufacture become common place as a method for making things, then, as in the desktop publishing revolution, there will need for systems which will allow the wider public to make use of the possibilities that these production technologies afford. The systems proposed in this project can be seen as embryonic versions of such systems, in which anyone can engage in a process that sits somewhere between a game and a CAD tool to generate one-off forms.

5. REFERENCES

[1]   Gullichsen, E.; Chang, E.: Generative design in architecture using an expert system, The Visual Computer, 1(3), 1985.
[2]   http://www.fluidforms.at/shop/catalog/index.html
[3]   Unver, E.; Dean, L.; Atkinson, P.: 'FUTURE FACTORIES': Developing Individualised Production Methods, Proceedings of 3rd International Conference on AED Advanced Engineering Design, Prague, Czech Republic, No 216, ISBN 8086059359, 2003
[4]   Marshall, J.; Unver, E.; Atkinson, P.: Automake: Generative systems, digital manufacture and craft production, GA2007, the 10th Generative Art Conference., Milan, Italy, Proceedings to be published, 2007.
[5]   http://www.georgehart.com
[6]   Dormer, P.: The Salon de refuse? in Dormer, P (Ed) The Culture of Craft, Manchester, Manchester University Press, 7, 1997.
[7]   Francis, E. H. Tay *et al*: Distributed rapid prototyping- a framework for Internet prototyping and manufacturing, Integrated Manufacturing Systems, 12(6), 2001, 409-415.
[8]   Atkinson, P.: Post Industrial Manufacturing Systems: the impact of emerging technologies on design, craft and engineering processes, Proceedings of Challenging Craft, Aberdeen ISBN 190108583x, 2004.
[9]   www.virtools.com
[10]   Dean, L.T.; Atkinson, P.; Unver, E.: Evolving Individualised Consumer Products, Proceedings of the 6th European Academy of Design conference, Bremen, Germany, ISBN 3-89757-290-7, 2005.