

A pattern language for describing documents

Rob Waller and Judy Delin *April 2011*

To cite this paper:

Waller R and Delin J (2010).
Towards a pattern language
approach to document
description. Paper presented at
Multidisciplinary Approaches
to Discourse, Moissac,
France, 2010. Technical paper
4, Simplification Centre,
University of Reading.

Experienced designers can draw upon a set of solutions to common problems they meet, but, because they don't always have names for them, they are hard to discuss or teach. That's why pattern libraries have developed as a useful way to collect and describe good practice.

This conference paper introduces the concept of design patterns, and reflects on their relationship to document genres.

Towards a pattern language approach to document description

Robert Waller, Judy Delin

The Simplification Centre, University of Reading, UK
r.waller@reading.ac.uk
judy.delin@roedelin.com

Abstract

Pattern libraries, originating in architecture, are a common way to share design solutions in interaction design and software engineering. This paper introduces the approach, and explores its application to functional texts as a way of documenting common design problems along with their potential solutions. In particular, it seeks to place patterns in the context of genres, with each potentially belonging to a ‘home genre’ in which it originates and to which it makes an implicit intertextual reference intended to produce a particular reader response in the form of a reading strategy or interpretative stance.

Keywords: pattern, genre, layout, typography

1 Background and context

Information design is a relatively young discipline, which struggles with the lack of a usable descriptive framework. By usable, we mean one that can be used to teach or define effective design strategies that at present tend only to be known tacitly by experts. Examples of practical uses for a descriptive framework are when government regulators prescribe formats for consumer information, when publishers specify formats for textbooks, or when insurance companies set up standard styles for customer communications. So by a *usable* descriptive framework we mean, in effect, one that is to a degree prescriptive as well as descriptive.

Without anything analogous to ‘grammaticality’ to use as a yardstick, information design tends to rely instead on success measures that are harder to test, such as usability. In practice, rigorous testing with users is often impractical – and so practitioners rely more on ‘knowing what works’ from experience. Communicating this expertise, however, is not straightforward when no established descriptive framework is in place to distinguish between good and bad practice. Prescription, then, might allow us a means of judging – or at least some rules of thumb – through which we can be of use as trainers and designers in the practical world. And, in addition, the much newer field of corpus-based research on multimodal documents also lacks metrics for choosing what we should include, and what exclude, from corpora.

A number of frameworks have been proposed from within the study of typographic or information design (eg, Twyman 1979) but most aim only to be descriptive, classifying objects of analysis according to theoretical schemata. This is fine as far as it goes, but while these frameworks may help us to organise phenomena that we find, and understand the influences that underlie them, they are not intended to be the basis of the kind of practical guidance that we have argued is needed.

The programme described in this paper builds on genre-based approaches (eg, Bateman 2008; Delin, Bateman and Allen, 2002; Waller 1987, 1990), using the concept of pattern languages and pattern libraries – an approach that originates in architecture, but which has been fruitful in information design’s close neighbour, interaction design. However, while genre theorists have tended to focus on explaining discourse types that already have names, the compilation of a pattern library is to a large extent a naming exercise. As one commentator put it:

‘[Naming] is one of the real powers of ... patterns. They not only expose a solution but they give it a name. They create a classification system. They form a vocabulary, a language. They provide a way for people to talk about the concept and a way to recognize the solution when a similar problem context arises in the future.’ Scott (2006).

2 The origin of the pattern language approach

In this context, *pattern* refers not to repeating decorative effects (for example, on wallpaper), but to configurations found consistently within recurring design solutions to common problems. They may be patterns of words, visual configurations, or a combination of both.

The term *language* needs qualifying also – it is used loosely here and does not refer just to verbal language or discourse, but to any systematic relationship between elements of almost any kind. We use it by way of reference to its originator, the architect Christopher Alexander,

and in practical applications the more realistic term *pattern library* has become more common.

Christopher Alexander (1977, 1979) developed his pattern language to describe consistently observed solutions to common problems that he and his team found in a wide range of human settlements – it is a way of describing forms found in vernacular architecture that have evolved naturally in response to human needs, rather than out of theoretical models (and in particular modernist approaches).

The idea of patterns is fundamental to human thought, and is not, of course, original to Alexander. In communication theory, the definition of rhetorical patterns goes back to classical Greece, and the term is frequently used by linguists working at various different levels of analysis – particularly at the discourse level (eg, Hoey 1983, Hunston & Francis 2000). For information designers, Alexander’s pattern language approach is attractive because it lends itself to a prototypical rather than taxonomic approach, which corresponds closely to how design is traditionally taught and practised (but not necessarily articulated). Experienced practitioners of any art, trade or craft are often able to recognise problems they have met before, and to call on a repertoire of possible solutions. Pattern libraries are an attempt to make explicit these traditionally tacit repertoires, and require the involvement of ‘reflective practitioners’ (Schön 1983) as well as descriptive analysts and this is reflected in our project team.

A typical example of an Alexander pattern is COURTYARDS WHICH LIVE (pattern 115). A courtyard allows us to resolve our desire to be outdoors and our need for protection – what Alexander calls a ‘living courtyard’ includes paths that cross, an opening to a wider space and a sheltered porch. Without these things, the courtyard becomes claustrophobic, rarely visited, and neglected – a ‘dead courtyard’. Good spaces, created in this way, aim to achieve a quality which, having rejected as inadequate such terms as ‘alive’, ‘whole’, ‘comfortable’, ‘free’, ‘exact’, ‘egoless’, and ‘eternal’, Alexander calls ‘the quality which has no name’. Linguists might similarly reject terms such as ‘grammatical’ as only partially adequate to capture the qualities of a discourse segment that it is correctly formed, relevant, cohesive and so on – and which might therefore count as a ‘good’ discourse contribution.

In order to help us to build in this way, Alexander captures the characteristics of what he observes to be successful environments through a series of 253 patterns (Alexander 1977). The patterns are presented systematically, and it is this approach that has been taken up in fields outside architecture – in particular, by software engineers (Gamma et al 1994). In fact, while the idea of pattern language is little more than a footnote in its original context of architecture, it is now a mainstream approach in software engineering. Software engineers were attracted to the approach because they needed a way to organise a range, or library, of configurations for software objects, to make them accessible for engineers in need of a solution to a problem that another engineer might have previously encountered.

Interaction designers (eg Tidwell 1997, 2005) have also adopted this approach. In contrast to paper document users who are expected to spend long enough with each document to become used to its unique conventions, web users move quickly between different information environments and need them to behave consistently. So interaction design as a field has had to quickly evolve a consistent set of rules that developers can use, and that users can intuitively grasp, to ensure that user effort is focused on accessing content rather than figuring out functionality. Pattern libraries have proved to be a useful way for interaction designers to share best practice.

So for Alexander, and for followers in other disciplines, a pattern is a format for capturing insight into common problems and their solutions, and for understanding the relationships between higher and lower order patterns (from a city to a shelf). In this paper we consider whether it is also a useful format for capturing similar insight about documents.

3 How problems relate to solutions in pattern languages

In his book *A Pattern Language*, Alexander describes patterns thus:

‘The elements of this language are entities called patterns. Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of a solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.’ (Alexander *et al*, 1977:x).

We will take as examples two of Alexander’s patterns: the ENTRANCE ROOM (Pattern 130; Alexander *et al* 1977: 622) and the WAIST-HIGH SHELF (Pattern 201; Alexander 1977:922).

The patterns take the form of the statement of a problem or need, followed by a solution. In the case of WAIST-HIGH SHELF, the problem statement is as follows:

‘In every house and every workplace there is a daily ‘traffic’ of objects which are handled most. Unless such things are immediately at hand, the flow of life is awkward, full of mistakes; things are forgotten, misplaced.’

There is then a discussion of how the problem might be solved, followed by a summary (in bold) of the solution:

‘Build waist-high shelves around at last part of the main rooms where people live and work. Make them long, 9 to 15 inches deep, with shelves or cupboard underneath. Interrupt the shelf for seats, windows, and doors.’

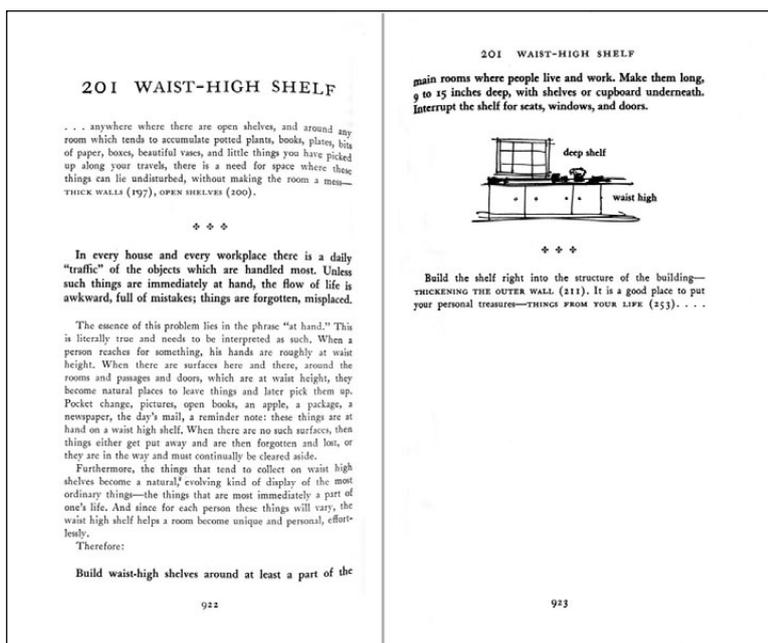


Figure 1: A typical spread from ‘A pattern language’ shows the key components: a title, an introduction that links to higher order patterns, a problem statement in bold, and an illustrated explanation.

There are two interesting things to note. One is that the pattern name in this case is the name of the solution – build a waist-high shelf. However, in the case of ENTRANCE ROOM, the solution is a particular design of entrance room, and the pattern name is more a general topic. Other pattern names are different again: SLEEPING IN PUBLIC, for example, recommends building outdoor environments that contain sheltered benches, away from traffic, where people can read the paper and doze off. In this case, the pattern is named after a habit or desirable activity. The one thing Alexander does not do is name the pattern after the problem: we don't see 'INHUMAN SCALE BUILDING', for example, or 'EVERYDAY OBJECTS OUT OF REACH'. This might be a clue to which part of the several pages that make up the pattern 'definition' is the actual 'pattern': an alternative description of pattern might be, it seems, 'loosely-specified design solution that solves a particular problem'. In the description given at the beginning, too, Alexander *et al* do suggest that patterns are both the problem *and* the solution, together.

The other interesting issue is that the pattern ENTRANCE ROOM actually contains a recommendation that there should be a waist-high shelf within the room. As point 4 of a 6-point series of recommendations, Alexander *et al* (p 624) suggest that there should be a 'shelf near the entrance' which is 'at about waist height', and provides further onward references to these and other patterns that are relevant to the satisfactory construction of the entrance room. This tells us that patterns, in his view, are recursive: it is quite normal for a pattern to contain 'calls' to several other patterns that are required to fulfil it. Whether this is full recursion or not we are not sure, but it does at least mean that patterns can be embedded within one another, in that the solution to one problem can invoke another pattern.

This makes sense, if we remember that problems can always be broken down into sub-problems, or goals into sub-goals, in computer planning terms. So, the problem 'make coffee' creates a sub-goal 'find coffee jar' which itself creates other sub-goals involving opening cupboards, and so on.

Going back to the summary of patterns as 'loosely-specified', it is clear from Alexander *et al*'s book that they must be so: if we 'can use this solution a million times over, without ever doing it the same way twice', there must be enough leeway in the solution to implement it in many different ways. There will be a big gap, then, between the notion of a pattern as intended by Alexander *et al*, and a notion of pattern that is implementable and computationally tractable.

4 Patterns in information design

To see whether patterns are a notion that is relevant to information design, we can look at an example of a relatively common problem in forms design: that of getting people to supply their phone number.

In Figure 2 we can see from the data collected by Crofts (2009) that there are a variety of ways of doing this even in a limited sample of four application forms. What is interesting about them is that they are more or less strongly constrained in terms of the format of the information the user can put into them.

Visa

4.2 Home (landline) telephone number

4.3 Mobile telephone number

Housing benefit

Your daytime phone number

Code Number

What is this number? Home Work Mobile Textphone

Please tick.

Child benefit

Daytime phone number

What is this number? Home Work Mobile

Tax

Daytime phone number including area code

Evening phone number including area code

Mobile phone number

Figure 2. Data from Crofts, K. (2009)

The tax form is the most constrained, in that it requires a separation of the digits into individual boxes and assumes a maximum of 14 digits. Housing benefit is the next most constrained, in that it divides the box into ‘Code’ and ‘Number’. The Visa and Child Benefit boxes are hardly constrained at all, in that they don’t suggest a format for the number or a maximum number of digits although they do employ different strategies for capturing what kind of phone number has been supplied.

We can see from this brief survey of solutions that not many people are ‘doing it the same way twice’. Some of the differences between solutions may not matter – they may be arbitrary side-effects of choices made at a different stage in the design process (for example, the choice of typeface or colour, and the thickness of lines around boxes). But some may matter in particular circumstances. For example, separate character boxes are often a sign that Optical character recognition (OCR) is being used to read the user’s data. Captioned sections (‘code’ + ‘number’) may be intended to prevent people missing out one part of the information requested. So a pattern definition needs to distinguish between its essential, or constituent features, as distinct from those that can remain accidental or contingent on other design imperatives (which might include features essential to a higher order pattern).

A question for the analyst is: looking at these samples, should we identify one loosely specified pattern, to be called PHONE NUMBER (after all, these are all reasonable ways of getting a phone number), with range of potential realisations as graphic elements, or should we identify three patterns (OPEN BOX, STRUCTURED BOX and OCR BOX), each of which has been applied to the topic of phone number, as distinct from, say, name, date or national insurance number?

5 How do patterns relate to genres?

Multi-modal studies of discourse have used genre as a key concept. Whatever else a genre may be, and however it is defined, it tends to be something that has already been given a name by its community of users: for example, leaflet, form, textbook, workshop manual, romantic novel, or crime novel. One of us has previously suggested that genre names evolve naturally, the arrival of a name signifying the achievement of communicative force by a new genre (Waller, 1987, page 285).

As we have already remarked, the identification of patterns is in one key respect the opposite of this – it is a deliberate naming exercise that recognises the existence of structures in documents that recur and are judged to be effective, but which have not acquired names naturally, except perhaps within a restricted community of practice (for example, within a particular studio, designers might refer to a layout where all items on a spread hang down from a common position, as a ‘washing line’, a term not shared by their readers). Pattern libraries articulate common solutions that designers use, so they can be shared and discussed.

Patterns are also distinct from genres because they are assumed to occur at various different levels of analysis, and many occur across multiple genres (that is, in documents which have very different purposes, content, format, context, etc). This was an explicit goal of Jenifer Tidwell, one of those responsible for introducing the pattern language concept to interaction design. Indeed, she saw pattern libraries as harnessing techniques not only from multiple genres but from multiple channels:

‘[A pattern language] would enable us to more methodically draw on expertise in related fields, such as book design, consumer electronics, the design of control panels (for cars, airplanes, power plants), video games, the Web and hypertext, and speech-driven interfaces.’ (Tidwell 1999)

A further distinction is that while the power of genres lies mostly in their adherence to convention, patterns may work not because they represent visual conventions that readers have learned, but because they represent other sources of communicative power. For example, they may represent good ‘gestalt’ – layouts that communicate connections, structures and separations by harnessing the natural tendencies of our perceptual systems to seek sense in visual form. Or they may work because they represent insight into the strategies and behaviours of typical readers.

6 Prototypes and peripheries

If there are some patterns that are most used, most familiar, or more constrained, or that are otherwise considered ‘best’ for a particular genre, we might think of those patterns as the prototypical elements of a genre. And similarly, those typographic and graphic solutions to the display of a pattern that normally work best can be thought of as prototypical solutions to a pattern.

The notion of prototype is inspired by Wittgenstein’s concept of family resemblances (Wittgenstein 1953) and developed by Rosch (1973; see also Taylor 2003). It accounts for the fact that humans tend to group things into classes for the purposes of convenient identification and understanding, and that some members of those classes may appear to be more ‘central’ members than others. For example, a penguin makes a worse prototypical bird than a robin or

a blackbird, because it can't fly and is an odd shape. The purpose of a prototype and the human ability to group things around it is basically because things in the real world differ from one another, but that some things (birds, chairs, cars, democracies) share enough common features for us to be able to identify them as instances of the 'same thing':

'The world consists of a virtually infinite number of discriminably different stimuli. One of the most basic functions of all organisms is the cutting up of the environment into classifications by which non-identical stimuli can be treated as equivalent.' (Rosch *et al*, 1976:383).

So peripheral members of a group are open to classification as part of more than one such group (for example a table lamp is a peripheral member of the categories 'furniture' and 'electrical household appliance'). In terms of document genres, then, we might think that there are forms that are 'formier' than others, and newspapers that are more newspaperly. By extension, there are elements of such documents – pattern solutions – that make more or less prototypical solutions to their problems.

For example, Figure 3 shows quite a good solution to the problem of eliciting a name on a form and is typical of the forms genre in its current state in the UK. Users of the form in Figure 4, however, often fail to supply the name correctly, because the sentence-completion solution used to elicit the name is now largely obsolete. The same solution seems quite at home, however, in the children's party invitation, which is a more peripheral member of the forms genre.

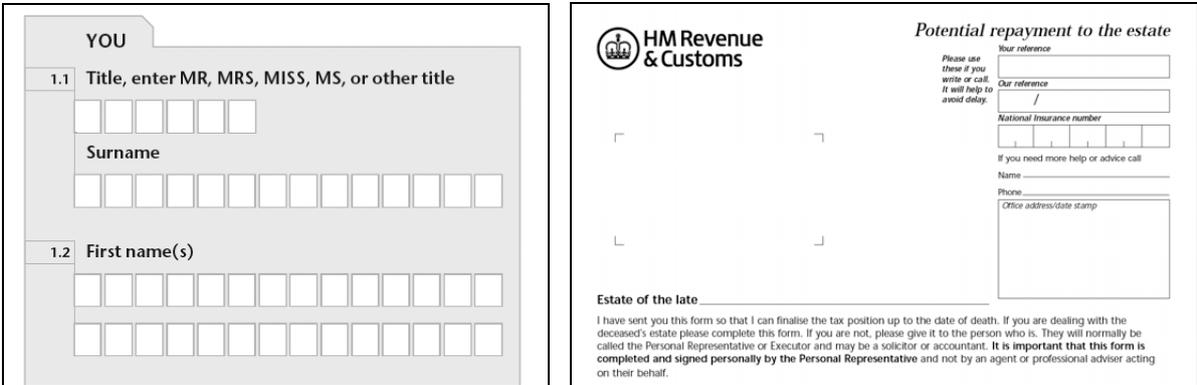


Figure 3: Two ways to elicit someone's name. The left-hand example, using the CHARACTER BOXES solution is more prototypical of the current state of the forms genre. The right-hand example uses the solution SENTENCE COMPLETION which is largely obsolete, and therefore peripheral.



Figure 4: The SENTENCE COMPLETION solution seems quite at home in this prototypical party invitation (making it also a peripheral member of the forms genre).

While patterns can occur within different genres, it may well be the case that many of them have a 'home genre' in which they are an essential feature. For example, the pattern LIST OF INGREDIENTS is an essential feature in its home genre 'recipe book', but it also occurs in the genre 'form' (where users might be given lists of key information to gather before starting).

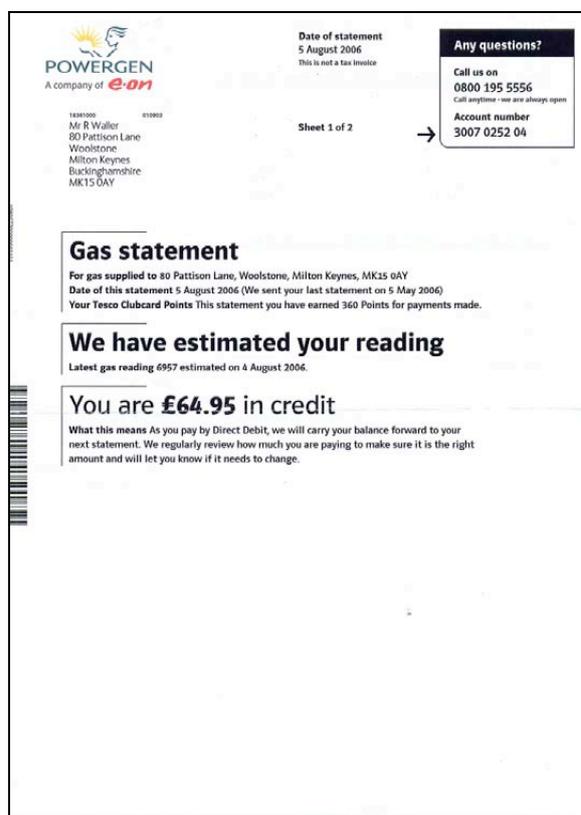


Figure 5: The NEWS HEADLINES pattern in its home genre (left) and in a gas bill (right)

Figure 5 shows how the pattern NEWS HEADLINES has been transported from its home genre, ‘newspaper’ to the genre ‘gas bill’. The resulting bill thus departs from its genre, but it nevertheless works because the headlines enable an effective reading strategy. Indeed, the use of headlines is an implicit intertextual reference to the newspaper genre, suggesting to readers that the reading strategy they use there (that is, a quick preview possibly, but not necessarily, followed by a detailed read of stories that interest them) is also appropriate for reading a bill. In time, if successful enough to imitate, the energy bill genre may shift.

While we are looking for a way to identify possible members of a set of solutions in a given pattern, therefore, we should note the following:

- Available solutions may be constrained by genre, but are also judged on their functionality in context, and the quality of their execution.
- Within the set created by the genre constraint, members will be more or less prototypical.

A hypothesis might be that solutions that are less prototypical might (a) be harder for users to identify visually as belonging to the pattern or the genre, and might therefore cause slower response rates and/or higher error rates, and (b) might, if they are less constrained, be more likely to turn up as possible solutions to other patterns. In this case, the more prototypical a solution is to pattern B, the more likely it is to cause confusion when used as a solution to pattern A – even if it appears within A’s set of reasonable possible solutions.

Cohen and Snowden (2008) have indeed demonstrated a correlation between the familiarity of document elements to readers and their performance in literacy tests. They use the term ‘document mental model’ to describe the kind of genre-specific knowledge required by competent readers that should be anticipated by competent document designers.

‘Readers are likely have a different mental model for each specific document type with which they are familiar. When confronted with a document, readers may recall and use these mental models, which, if accurate, should aid them in locating the vital information. For example, menus often contain the price of a dish to the right of the listing for that dish. For those with an accurate “menu” mental model, a request to locate price should be facilitated when the information is near the predicted location and inhibited when it is not.’ (page 19).

7 The research context of this discussion

The pattern language approach introduced in this paper is part of a wider research programme that includes, firstly, the building of a document corpus, so we can demonstrate the frequency of patterns within a particular domain (in the first instance, financial services documents), and, secondly, the testing of documents (selected to include patterns and genres of greater or lesser prototypicality, as well as other variables such as the strength of graphic and linguistic signalling) with users who come with different levels of experience and financial capability.

References

- ALEXANDER, C., ISHIKAWA, S., & SILVERSTEIN, M. (1977). *A pattern language. Towns. Buildings. Construction*. New York: Oxford University Press.
- ALEXANDER, C. (1979). *The timeless way of building*. New York: Oxford University Press.
- BATEMAN, J. (2008). *Multimodality and genre: a foundation for the systematic analysis of multimodal documents*. London: Palgrave Macmillan.
- COHEN, D. J., & SNOWDEN, J. L. (2008). The relations between document familiarity, frequency, and prevalence and document literacy performance among adult readers. *Reading Research Quarterly*, 43(1), 9-26.
- CROFTS, K. (2009) *Companion report: Patterns and Problems*. Unpublished project report for MA in Information Design, Department of Typography and Graphic Communication, University of Reading, UK.
- DELIN, J., BATEMAN, J., & ALLEN, P. (2002). A model of genre in document layout. *Information Design Journal*, 11(1), 54-66.
- GAMMA, E., HELM, R., JOHNSON, R., & VLISSIDES, J. M. (1994). *Design patterns: elements of reusable object-oriented software*. Addison-Wesley.
- HOEY, M. (1983). *On the surface of discourse* London: Allen & Unwin.
- HUNSTON, S., & FRANCIS, G. (2000). *Pattern grammar: a corpus-driven approach to the lexical grammar of English*. Amsterdam: John Benjamins.
- ROSCH, E. H. (1973). Natural categories. *Cognitive Psychology*, 4, 328-350.
- ROSCH, E. H., MERVIS, B., GRAY, W., JOHNSON, D., & BAYES-BRAEM, P. (1976). Basic objects in natural categories. *Cognitive Psychology* 8, 382-439.
- SCHÖN, D. A. (1983). *The reflective practitioner: how professionals think in action*. London: Temple Smith.
- SCOTT, BILL (2006) contribution to a web discussion on pattern libraries: *Design Patterns: Part 3* (<http://www.lukew.com/ff/entry.asp?350>)
- TAYLOR, J. R. (2003). *Linguistic categorization*. Oxford University Press.
- TIDWELL, J. (1997). *Common ground: a pattern language for human-computer interface design*, from http://www.mit.edu/~jtidwell/common_ground.html
- TIDWELL, J. (2005). *Designing interfaces: patterns for effective interaction design*. O'Reilly.
- TWYMAN, M. (1979) A schema for the study of graphic language. In *The Processing of Visible Language*, (Eds, Kolers, P.A., Wrolstad, M.E. & Bouma, H.) Plenum, New York

WALLER, ROBERT (1987). *The typographic contribution to language: towards a model of typographic genres and their underlying structures*. Unpublished PhD thesis, Department of Typography and Graphic Communication, University of Reading, August 1987.

WALLER, ROBERT (1990) 'Typography and discourse', in R Barr et al (eds), *Handbook of Reading Research*, vol II, New York: Longman, 341–380.

WITTGENSTEIN, L. (1953). *Philosophical investigations*. Oxford: Basil Blackwell.