MODELING REGULATIONS AND INTENTIONS FOR URBAN DEVELOPMENT: The Role of Computer Simulation in the Urban Design Studio

Josep Maria Fargas and Pegor Papazian Research Affiliates,
Massachusetts Institute of Technology

Abstract

In this paper we present a strategy for modeling urban development in order to study the role of urban regulations and policies in the transformation of cities. We also suggest a methodology for using computer models as experimental tools in the urban design studio in order to make explicit the factors involved in shaping cities, and for the automatic visualization of projected development. The structure of the proposed model is based on different modules which represent, on the one hand, the rules regulating the physical growth of a city and, on the other hand, heuristics corresponding to different interests such as Real Estate Developers, City Hall Planners, Advocacy and Community Groups, and so on. Here we present a case study dealing with the Boston Redevelopment Authority zoning code for the Midtown Cultural District of Boston. We introduce a computer program which develops the district, adopting a particular point of view regarding urban regulation. We then generalize the notion of this type of computer modeling and simulation, and draw some conclusions about its possible uses in the teaching and practice of design.

Knowledge and urban regulations

Most urban regulations are formal descriptions of a set of generalized building types. These regulations are motivated by some criteria such as the fair and sanitary distribution of natural light, limiting population density, and the like [Quarter, 1988]. Unfortunately, most of these criteria are not explicitly reflected in the norms which they inspire, and urban regulations allow only a limited range of possible implementations of the criteria, namely the ones which can be expressed in the vocabulary of some common building types. One of the contributions that computer science can make to the field of urban design and planning is to increase the level of sophistication and generality in the expression of regulations. Not only is computer science a field in which issues of representation and the formal expression of abstract constraints are well-studied, but the computer has often been used as a tool for building models for simulation and testing of systems very similar to the ones treated by urban regulations.

In this paper we will present a first approach to the idea of using computer tools for the task of understanding and explaining urban development and its regulation. The urban design theory implicit in our treatment of the problem and in the modeling system which we will present stresses the importance of the conflict of different interests and forces which shape a city, rather than an approach to urban design as a formal problem. Our aim will be to capture design knowledge in the mechanism for regulating urban development such that the forces and criteria acting upon the transformation of cities can be readily observed in the system, resulting in a powerful pedagogical tool. The system we will present below is an implementation of one aspect of a general-purpose modeling shell based on the interaction of different modules. Each module represents what we have called a "metaphor" [Fargas and Papazian, 1992] or a particular way of seeing the environment and a specific interest in developing and transforming the physical model in question. In the case of urban development, one module could represent the perspective of a community group, another module might capture the point of view of a city-hall planner, yet another that of a certain kind of developer, and so on.

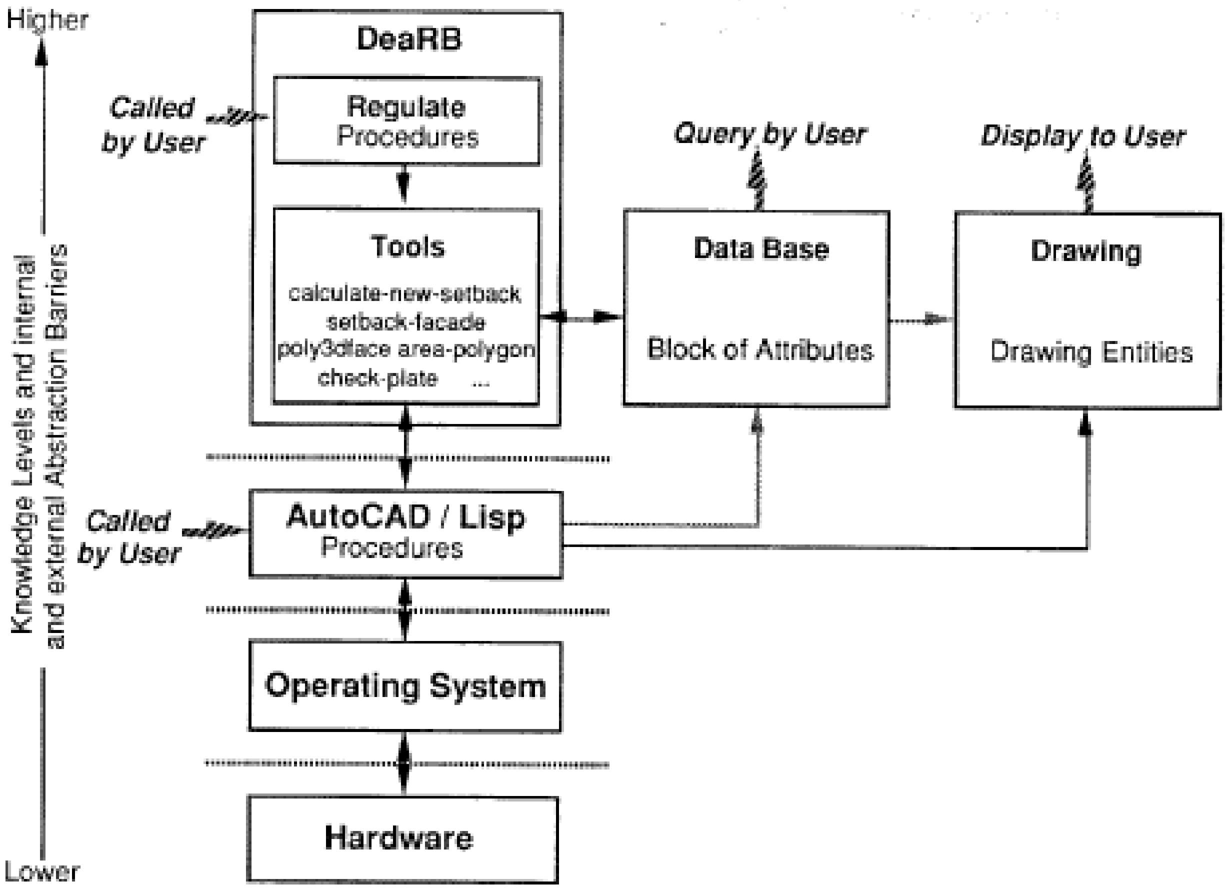
Designing Rules, Boston (DeaRB) is a modeling system which implements the Boston Redevelopment Authority (BRA) Downtown Zoning Code for the Midtown Cultural District (MCD) of Boston [Fargas, 1991]. The aim behind DeaRB is not to provide a complete zoning tool but to exemplify a method of using computers in the teaching and practice of design. DeaRB also illustrates how regulations as a set of rules influence design and suggests a way of designing rules which will promote "better" designs, that is designs that better express the purpose of the regulations and the design knowledge they represent.

Levels of abstraction

To implement regulations or rules about design, several assumptions and model scenarios have to be stated so as to be able to build from them. Regulations have to be abstract enough to encompass different situations under which the real solutions or instantiations of the rules might vary substantially [Lai, 1988]. Since regulations are very often designed based on specific design problems and generalized a posteriori, there is always a margin of "error" or inaccuracy related to very specific real situations in which rules do not apply properly. In these real life cases negotiation is very often the way out of the problem. In the case of designing a program that will implement regulations one has to state very clearly the frame of action and the limitations of the applicability of the rules. The person building the program can always design the "regulator" so as to ask the users what they want to do in a "non computable" situation.

Figure 1 illustrates the different levels of knowledge associated with abstraction barriers between platforms and, inside DeaRB, between procedures. It also shows DeaRB's interface strategy. There are two levels of decision that are relevant when designing a "regulator". The first is about the set of rules to replicate or to implement, that is the kind of designs the program is going to produce and the criteria according to which it will produce them. The second level of decisions is more intrinsically related to the implementation of the rules themselves. The regulated "object(s)" have to be studied so as to define what will be the set of topological and geometrical relationships that the program will have to know about so as to be able to relate objects and their parts to take the appropriate design decisions. This second level is decisive in the context of reaching a relatively "complete" system.

Figure 1
Diagram of DeaRB's relationship with the different platforms and levels of knowledge



When designing the program itself it is very difficult to encompass all the possible exceptional cases that one wishes to make the program responsible for and it is especially important to design the program with modularity or using object oriented techniques so as to be able to modify it easily when these cases appear.

Let us now attempt to see what DeaRB "knows". The program knows things related to different "levels of abstraction" and its body of knowledge is stratified according to different "abstraction barriers" [Ableson and Sussman, 1985]. This means that the same way the user does not need to know how the program works internally to be able to use it, the programmer does not need to know specific things about the computer or the programming language itself, such as how the processor internally binds values to specific parameters, etc., so as to be able to design a program to do specific tasks. The two levels of decisions that should be made when designing a "regulator" which we mentioned above are directly related to the concept of abstraction barriers and to the specific kinds of knowledge embedded in the program.

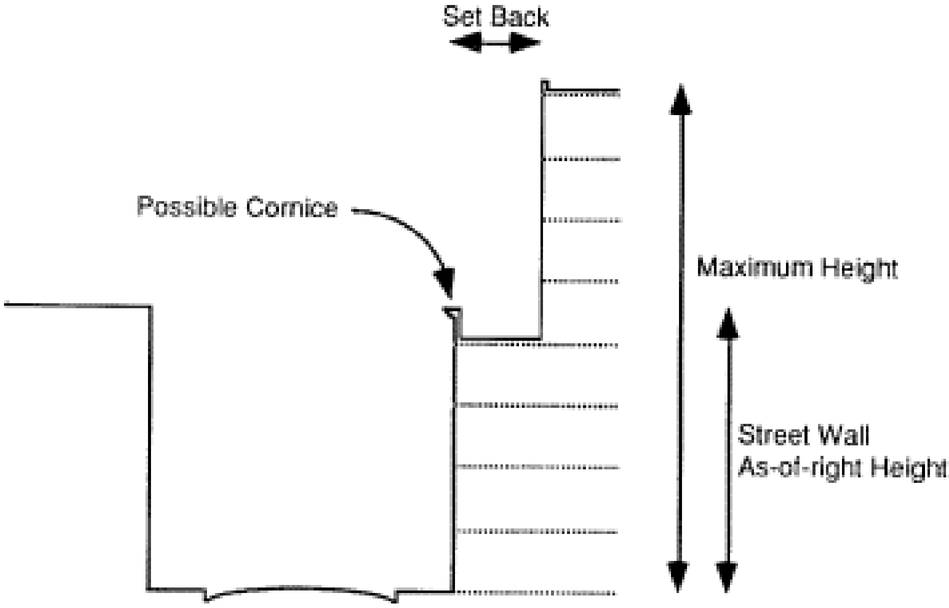
In relation to the first level of decisions, DeaRB knows the MCD Downtown Zoning code -as determined by the BRA's regulations [BRA, 1989]- which is expressed as a set of conditional rules applied before regulating any lot, and which is used by the system to derive building forms. The program also knows the zoning parameters of each lot such as the as-of-tight height limit, maximum height limit, set back from street wall, as-of-right floor-to-area ratio (F.A.R.), and maximum F.A.R., and, more importantly, how these parameters relate to each other (sec figure 2). DeaRB also knows how to build according to these parameters. Zoning parameters by themselves are not enough to derive the building envelope. Specific decisions need to be taken for instance in relation to floor heights, use and mix of uses, criteria about maximizing the built area, etc.

When designing a program like DeaRB there is a need to define a perspective from which to take design decisions from the point of view of each rule set. For example, to better understand a specific situation one can analyze it under different perspectives, the one of a rational planner, a traditional designer-planner, an advocacy planner, an economist-planner, etc. In DeaRB a basic perspective is the one we have called the "unscrupulous developer" because it tries to optimize the number of office floor in relation to residential floors, but without loosing any F.A.R., the specific details of the implementation will be explained below. All of these considerations are proper to a specific professional domain -the discipline of planning- and these domains usually build from either "lower level" knowledge domains or complementary or basic domains such as social science, economics, mathematics, geometry, etc. It is in this sense that when writing a program for designing, it is not enough to develop the specific domain of interest. One also has to build in lower level domains, for instance the geometrical, topological and logical domains. The more design-oriented CAD packages become, the less designers will have to build in these "lower level" domains. Hopefully future CAD packages will contain more sophisticated levels of abstraction on which to build. The question will then be what the appropriate abstractions are to build into a CAD system. It is in this sense that the details of the second set of decisions embedded in DeaRB becomes important.

In relation to the second set of decisions related to the implementation of the rules themselves or the regulations, DeaRB knows how to associate a specific drawing entity (or AutoCAD polyline) to its specific information database (or AutoCAD block of attributes) and vice versa. It also knows how to triangulate a polygon so as to be able to build surfaces (AutoCAD 3D faces) indispensable for producing rendered images, how to set back the facade of a lot in both cases where lots have neighboring walls and when they are isolated, how to find the center of gravity of a polygon and how and where to draw entities (polylines and 3D faces).

DeaRB assumes that the concept of "street wall" has highest priority, consequently, it will always build up to the as-of-right height limit, but if a set back plate is to be built, then it knows that it can start the plate lower than the as-of-right height limit, as if there were a "cornice" higher than the last floor of the lot from which the new set back plate will start (see figure 2). This kind of knowledge might remain a function of the "perspective" that in this case is directed towards optimizing the number of floors built while maintaining the cornice or strect wall height, as if the city had a "ceiling".

Figure 2
DeaRB's internal representation of the street cross section using all zoning parameters



Specifically in relation to the BRA regulations, DeaRB knows that a given facade of a lot might not be a strait line, thus it knows how to recalculate the new set back of the lot to match the desired area for the upper set back floors. This last kind of knowledge is somehow higher level than the rest since it builds upon other items of knowledge and has a higher degree of sophistication.
Finally, there are other kinds of things that DeaRB knows that are even lower level than the ones explained above. For example, it knows the heights of the different floors (ground floor, office floor and residential floor) that can be modified for each lot according to its use. DeaRB also knows, for each lot, the nature and location of the facade, whether it is just a line or a succession of segments and whether the lot is isolated and has no neighboring lots. In relation to the database, DeaRB knows how much area is built for each lot and this number can be later managed independently of the lots. This allows the execution of some statistics about built area in the MCD and the possibility of transfers of air rights. For example for historical buildings it calculates how much F.A.R. is not built so as to be able to sell it or use it in other lots by the same owner or the BRA itself. The underlying metaphor would be to keep the city with a "deformable" ceiling rather than a fixed one, preserving built volume over the MCD. It might even be possible to "export" air rights to non-equivalent areas establishing compensation factors such as price of land to maintain a fair exchange.

Range of problems handled by DeaRB

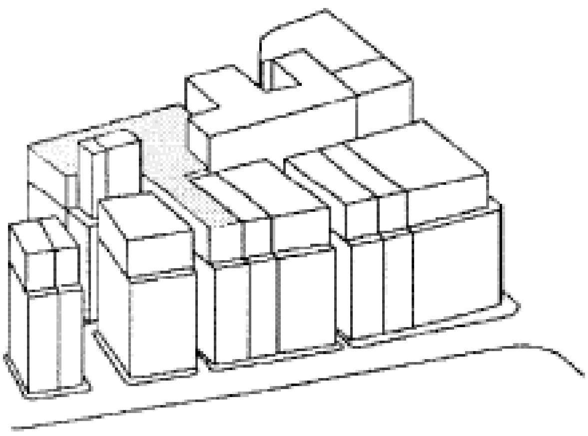
DeaRB handles only five zoning parameters of the MCD Downtown Zoning Plan and it knows how they interact (as-of-right height limit, maximum height limit, set back from street wall, as-of-right F.A.R., and maximum F.A.R.). These are the most significant zoning parameters. Other specific factors were not considered in order to keep the design problem simple. For instance, DeaRB does not know the regulations about shadows and their impact over the Boston Common and the Public Gardens.

In relation to the zoning parameters handled, DeaRB only allows for one set back although regulations allow several set backs in function of height and for specific cases. This was not implemented because of simplicity considerations and because it only applies to very few lots. The program can only handle two heights associated to two floor-to-area ratios, as-of-right and maximum, although there might be some cases in which depending on negotiations, a greater height might be allowed as well as a higher F.A.R than the maximum ones. In addition, there are no use considerations, which means that there is no possibility of specifying bonuses or limitations of zoning parameters according to use, as the Zoning code foresees. DeaRB also handles the maximum floor plates defined in the Zoning code (25,000 square feet for the MCD, section 38-19) and minimum floor plates (5,000 square feet) which were introduced to avoid impossible plates. Note that in the output produced by DeaRB, some lots do not have set back plates since their area is less than the minimum, although a developer could buy adjacent lots so as to have a higher lot area and to be able to build a set back plate.

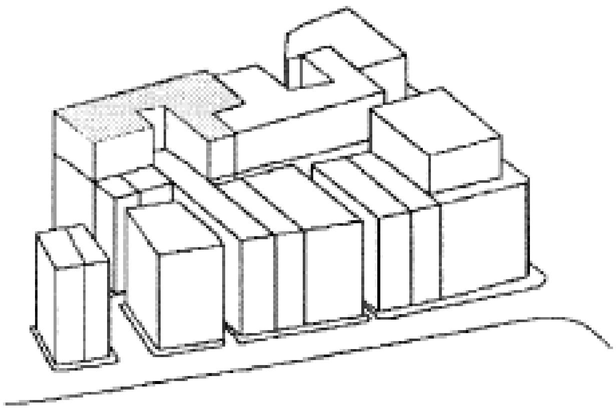
In its current implementation, DeaRB can handle most lots. As mentioned above, lots might have different kinds of facades (a strait line in neighboring lots, several segments stepped back or with indentations) and they might be isolated blocks. What DeaRB does not handle is the few cases in which a lot has two different facades, for example facing a street on one side and a back street on the opposite side, or special cases in which a lot has access rights to a main street through a kind of "corridor" but where the main area of the lot is in the back of the block, like Astor Theater in Tremont Street (see the right side picture of figure 3 in which the plate has a pronounced indentation due to the facade).

Figure 3
Screen bit maps of DeaRB's regulatory work on the Astor Theater block

"rough envelope"
Only heights limit applied



"maximum envelope"
All zoning parameters applied

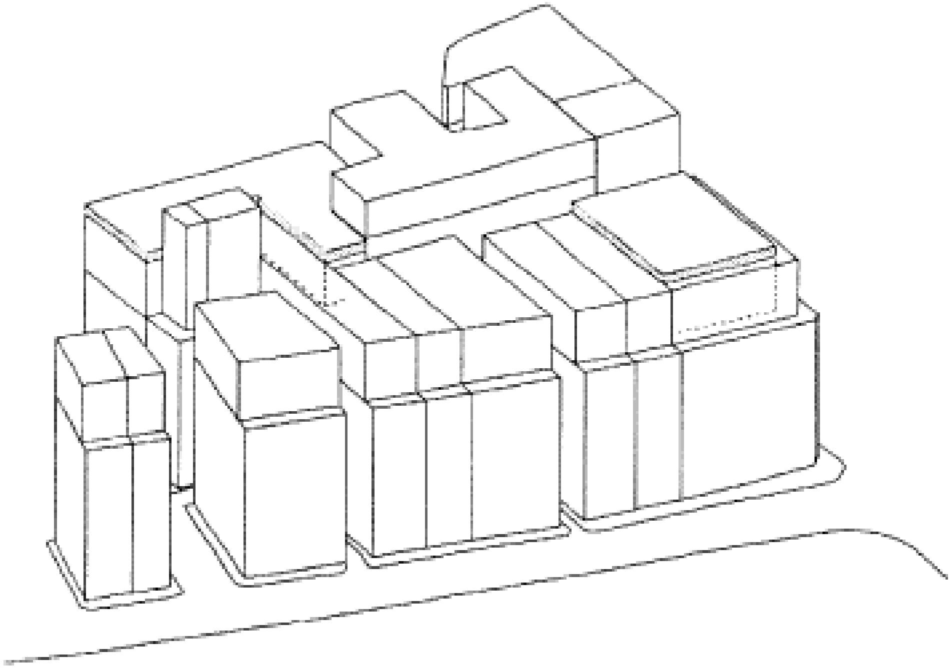


In these eases, when DeaRB tries to regulate the lot, it sets back only the portion of the facade to the street and tries to recalculate the new set back in function of the new area of the plate only moving the facade segment of the lot giving a non-satisfactory solution. Note also that if the lot has two noncontiguous facades then the more restrictive regulations are applied and this is why Astor Theater has a height lower than its neighboring lots in the back part of the lot, corresponding to the rear facade. As mentioned above, DeaRB does not handle this situations satisfactorily.

DeaRB can also build the new plate according to the specified area when possible. That is, if the maximum height allows to have a higher building when using the maximum plate allowed by the regulations (25,000 square feet), it is possible to tell DeaRB to build a smaller plate just by reducing the maximum area of the new plate. This is a useful feature since it makes it possible to compare solutions and decide which one to use. What DeaRB does not allow is to build several slender towers on top of a block. This might be included in other versions of DeaRB. Note that in the above example, in the picture on the right the regulated buildings are not built up to the maximum height because the plates are smaller than the minimum plate defined as 5,000 square feet.

Another feature of DeaRB is that when calculating the new F.A.R. for the set back plate, if there is still enough height, it rounds the F.A.R. to the next integer and considers it as the number of floors for the set back plate or "tower" so as to build all of the floors with the same area. This means that sometimes DeaRB will propose plates higher than the limit height by an amount always lower than the height of the minimal residential floor (see figure 4 and the marked lots).

Figure 4
Super-imposition of "rough envelope" and "maximum envelope" applied to the Astor Theater block. Note the marked areas in which heights are greater than those allowed by the regulations only by an amount lower than the lowest floor height (residential floor).



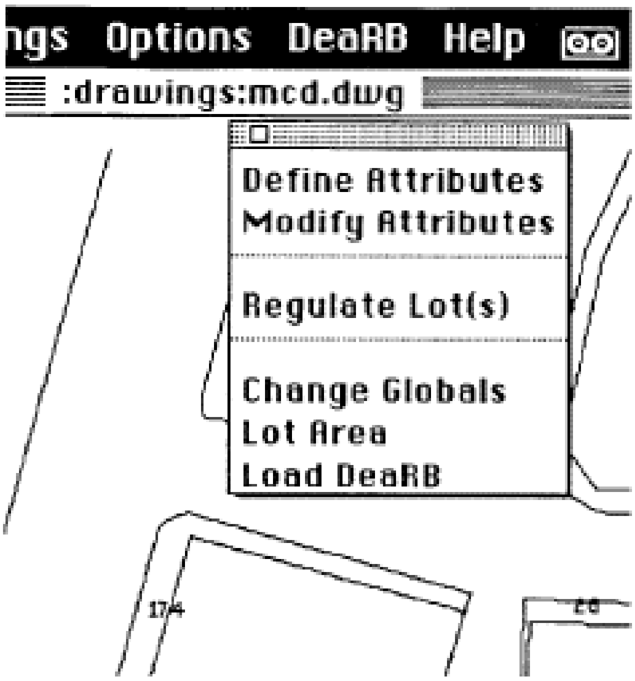
DeaRB considers that since the set back would be bigger than the minimum zoning set back there would be the possibility of negotiation with the BRA, specially in the case of an unscrupulous developer. Since this is applicable to all situations, in the case of isolated blocks the amount of facade is so big in relation to the block that the different set back is too small to get a "bonus" height. It seems unfair to allow such set backs in these cases but DeaRB allows them. The solution might be to implement some rules that consider street ratios or proportions, as we will explain below.

It is also possible to derive regulators by first designing a specific lot manually and afterwards, through a process of trial and error, deriving the zoning parameters for regulating the lot. The way DeaRB handles this is by changing the values of the block of attributes which might not be the best way since it is not very intuitive. We should point out that BRA's Zoning code states that the minimum set back for the MCD has to be at least 10 feet and the as-of-right height 90 feet when these are not specified. This means that there might be cases in which set backs do not make much sense, specially in lots with fronts in back streets, and with facades with considerable indentations. This is particularly important for the lot next to Astor Theater shown in the above pictures, and illustrates a point of vagueness in the regulations.

How DeaRB works

DeaRB is fully menu driven. The first thing to do when using it is to draw a lot as a simple AutoCAD polyline. The following functions of DeaRB which we will describe are the ones shown in the above image of AutoCAD's pull-down and detachable menus under "DeaRB" item of AutoCAD's menu bar (see figure 5). Once the lots to regulate are identified, DeaRB needs to be loaded (by clicking "Load DeaRB") if it has not been loaded automatically at startup.

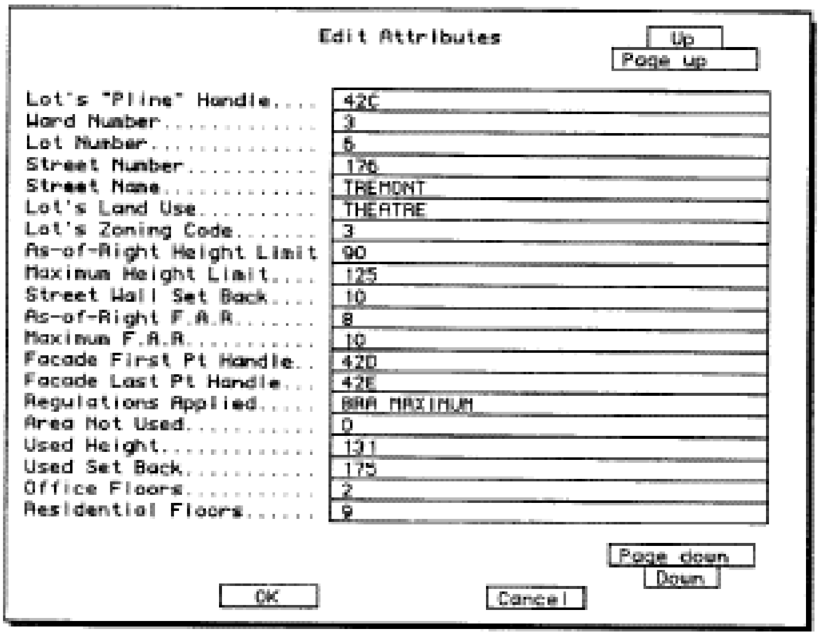
Figure 5
Functions of DeaRB's detachable pull-down menu under item "DeaRB" of the AutoCAD menu bar



The next step is to associate a block of attributes to each lot (by clicking "Define Attributes"), and finally the lot(s) can be regulated (by clicking "Regulate Lot(s)"). In the following explanation we will call "the user" any person who might work with DeaRB since it will need some external input to regulate and there are different ways to get back the new regulated parameters in function of the needs.

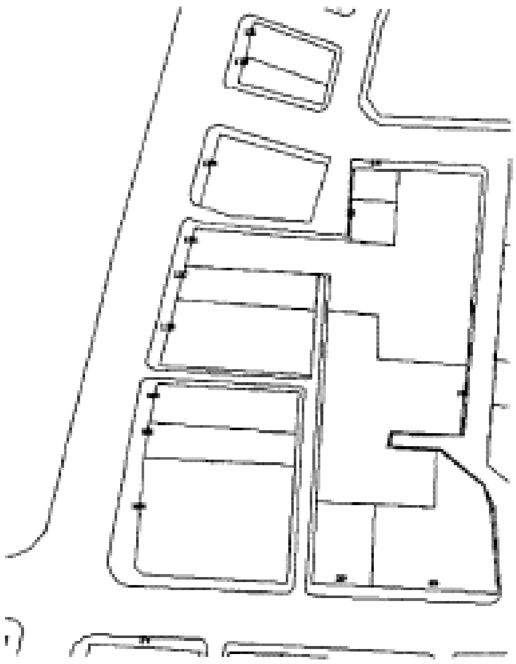
Before a lot can be regulated its list of attributes needs to be defined. The procedure LOTATT binds drawing entities and blocks through AutoCAD handles. It is invoked by clicking "Define Attributes" under the "DeaRB" item of the menu bar. DeaRB only prompts the user to input the parameter that define the lot, since it already knows the zoning parameters according to the zoning code. Once the parameter are fixed, there is always the possibility of editing them using the procedure MODATT (the procedure is invoked by clicking "Modify Attributes"). This procedure is specially useful for getting back the values proposed by DeaRB after regulating a lot, since there are fields specially designed to store the parameters DeaRB calculates (used height, potentially built area not used, number of office and residential floors, and the type of regulation set applied -i.e. BRA-) and these are only filled in after a regulation set is applied. DeaRB's pull-down menu also includes two additional functions. "Change Globals" changes the global parameters (first floor, office and residential floor heights, and maximum and minimum plates -although in the case of BRA regulations maximum plates are fixed). "Lot Area" returns the area of the lot picked by the user.

Figure 6
AutoCAD's representaton of the block of attributes for the Astor Theater lot



The screen image above (figure 6) is an example of the pop-up menu of DeaRB's AutoCAD block of attributes (LOTATTS) once these are associated to a lot by clicking "Define Attributes" as described above. The pop-up menu is brought up when the user selects the "Modify Attributes" item of the pull-down menu which internally calls procedure MODATT. The image below (figure 7) is part of the MCD computer model produced in AutoCAD. It corresponds to Astor Theater block used here as an example to illustrate how DeaRB works. Note that from the above list of attributes for the Astor Theater block, only the street number of the lot is "visible" because DeaRB uses it when it needs the user to refer to a specific lot.

Figure 7
The Astor Theater block in the Midtown Cultural District computer model

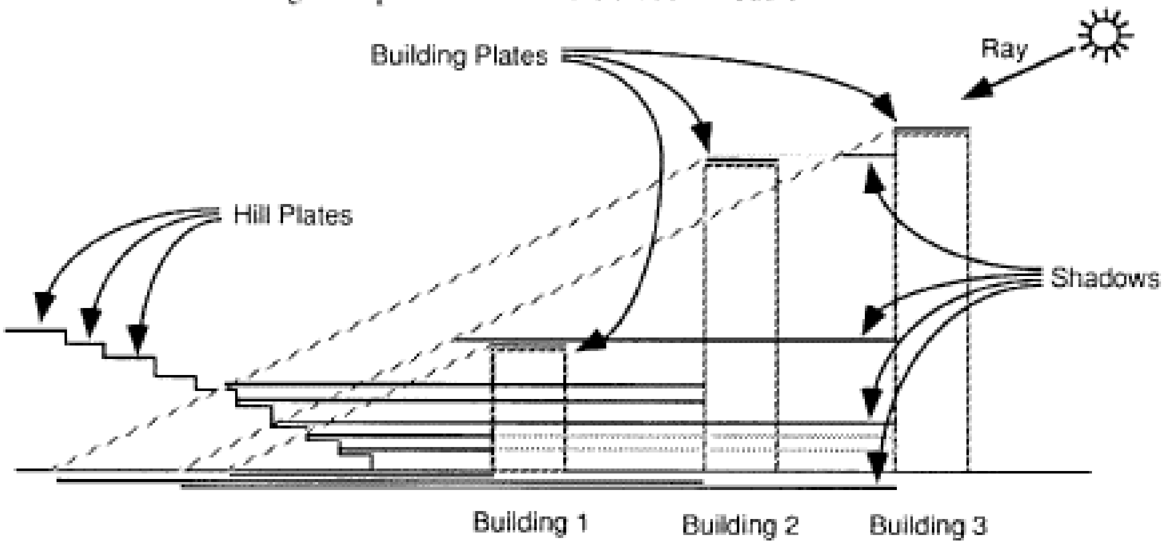


DeaRB has a general procedure, REGULATE, that when called (by clicking "Regulate Lot(s)" in the pull-down menu) prompts users on how they want to regulate the lot(s) -which regulations to use-, and from what perspective. We will use the BRA's set of regulations (MCD Downtown Zoning Plan) and the "unscrupulous developer" perspective to show how DeaRB works and how it manages its assumptions.

DeaRB always assigns special uses to the ground floor (commercial, cultural, leisure, etc.) whose height impacts the regulated design. Thus, DeaRB only knows about office and residential floor use (although one could fool the program and make it regulate other uses without DeaRB realizing it, since the knowledge of the program only refers to relationships between offices and residences).

In future versions of DeaRB there will be a module which will consider maximum shadows over the Boston Common and Public Gardens. As the MCD Downtown Zoning Plan says in section 38-16, "Each Proposed Project shall be arranged and designed in a way to assure that it does not cast shadows for more than two hours from 8:00 a.m. through 2:30 p.m., on any day from March 21 through October 21, in any calendar year, on any single Shadow Impact Area". There are two things that a shadow module will have to check: the first is the shadow itself and the second is the so called "ghost" effect. Due to the "ghost" effect a building might benefit from another existing building by hiding its shadow under that of the existing building. In this sense, a shadow module might be interesting not only to check shadow impacts, but even to "transfer shadow rights" and also to allow deriving a rough shape for the building from a given shadow. Figure 8 illustrates a possible implementation of the shadow module.

Figure 8
Cross section illustrating an implementation of the shadow module



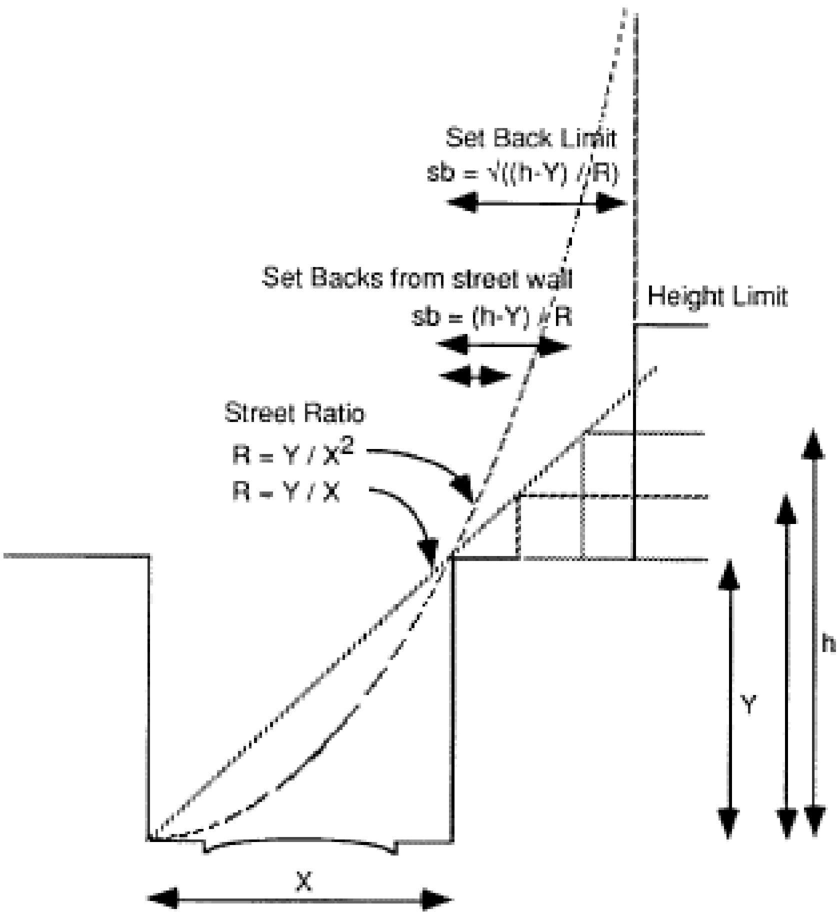
A heuristic approach to regulations

This section will expose the kind of knowledge that can be implemented using what we have called a "heuristics" approach. We do not pretend to give a complete set of rules, but just an alternative or complementary view of the zoning parameters implemented in the BRA regulations. The first heuristics we would like to introduce are based on the concept of street ratio to define what is a good street in terms of proportion, view [Lynch and Hack, 1984] and light (see figure 9). This concept was introduced during the development of the International Style and became a standard for designing streets or corridor-s considering light and health issues. In the MCD, since the street wall is defined at 90 feet and the mean width of the district streets is around 60 feet, the street ratio proposed by the regulations will be around 1.5. The way the regulations are written it is difficult to say if the street wall height refers to the concept of street ratio or to a purely formal consideration, like keeping a kind of ceiling at the old height of building cornices. This might depend on the site and, it might seem realistic to consider a set back limit after which the street ratio will not be applicable so as to give rise to more possible scenarios. This is specially relevant in the case of slender but high towers.

There should be the possibility of defining successive set backs according to either street ratios or fixed parameters for each lot or in function of the zoning code. Also, these set backs do not have to respond necessarily to the same street ratio, and a parameter can be introduced in the set back formula so as to account for this (sb = (h-Y) / R x parameter, R = Y / X), or the formula might change to an exponential curve street ratio (i.e., sb = √((h-Y) / R), R = y X2) in which the limit of the curve will he the set back limit.

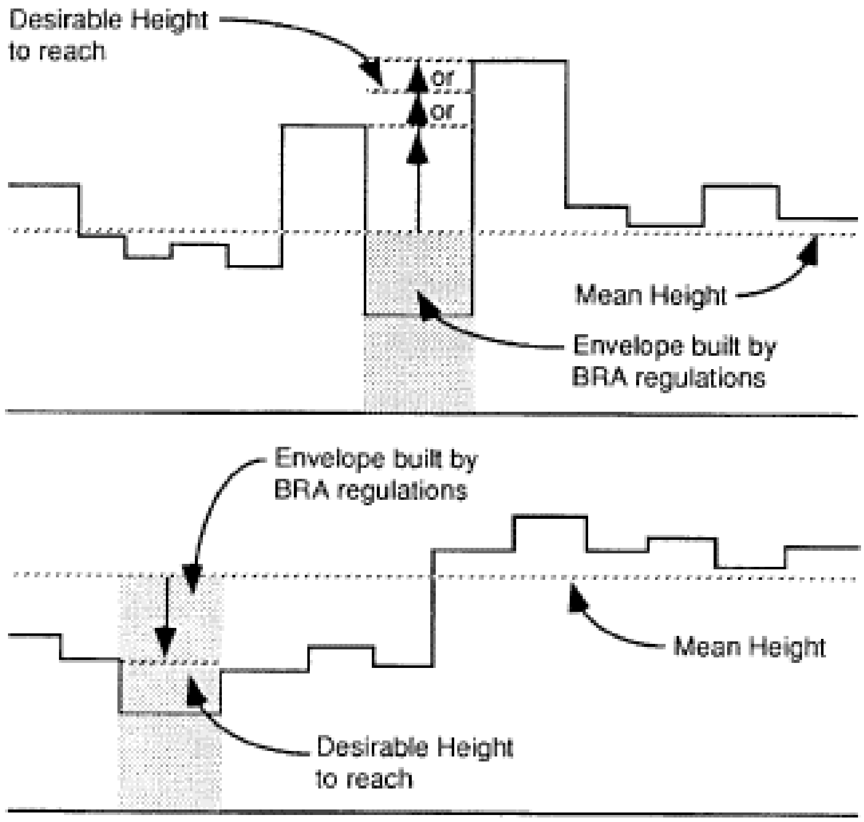
The exponential formula can be very easily implemented as figure 9 shows, and it has the advantage that there will not be any need of setting limit parameter since they will be implicit in the formula itself.

Figure 9
Street Ratio illustration



Besides what DeaRB already knows about floor heights, uses of the floors and BRA regulations, contextual information about neighboring lots will be fundamental in considering other buildings when regulating a lot. A basic criticism of the current regulations is that since they do not consider what is already built, in cases where buildings are higher or lower than the heights specified by the regulations, the concept of street wall does not stand: either buildings are too low and neighboring walls are only seen from the public space or buildings are too high in relation to the existing ones which are adjacent to them.

Figure 10
Cases in which The MCD Downtown Zoning Plan fails to give adequate solutions (street elevation)



One way height limits are derived is by using the mean height of a block, street, or district. This seems too rough to deal with the kinds of cases described above and which in practice are usually resolved by negotiations with the BRA (see figure 10). What we propose for future versions of DeaRB is to build some heuristics which it would use to decide up to what height to build considering context buildings, the zoning code and the above mentioned F.A.R. transfers.

Uses of the model in the study of urban design

The kind of system we have described can be used as an urban design laboratory, where ideas about the development of cities can be elaborated, tested and observed as in the example of DeaRB. In the context of an urban design course or studio, the effect of different policies governing the growth and transformation of cities are dealt with by referring to examples of existing urban fabrics. Often, the result of such an approach is that formal considerations which can be readily illustrated by traditional means take precedence over other factors which are better studied analytically. With a computerbased modeling and simulation system, applications such as the one described above can be used to perform a variety of exercises which will help improve one's understanding of urban development phenomena. One application of such a tool in the context of a workshop, for example, would first require the participants to develop a set of basic formal primitives which will form the urban design vocabulary of the system, and a set of criteria regarding the desired relationships and characteristics of those primitives. After this group effort, the exercise would continue with the adoption of two kinds of roles by different parties within the workshop. Those adopting the first role would impose certain restrictions constraining the introduction of the primitive elements (corresponding to zoning regulations for example) while each participant in the second role would develop certain systematic strategies for the introduction and transformation of the primitives, trying to bring about specific properties in the evolving model. The exercise can have a very simplistic character, evolving in an abstract context or, like the DeaRB system, it can have a level of realism which will allow comparisons with actual urban contexts. Indeed the exercise could consist of an attempt to partially model an existing city, trying to discover the different development behaviors which have interacted with the cities urban regulations.

At a more technical level, the kinds of applications of computer, to urban design proposed in this paper can he considered examples which go beyond simple tools for increasing productivity, performing visualization or even studying languages of forms (although each of these forms part of such applications). The work we have presented includes different levels of technical issues, ranging from the customization of a commercially available CAD program, to the development of representations appropriate to the type and scale of the problem at hand. But beyond these issues, each of which could form the basis of a class specializing in computer applications, the modeling system illustrates a use of computers which not only attempts to take full advantage of the characteristics of the technology, but which is also oriented specifically towards the practice of urban design and planning.

At another level, an already elaborated system, such as a more advanced version of DeaRB, or systems resulting from workshops such as the one described above can be used to study issues such as conflict resolution and urban policy making. Similarly, enhancements of the model can be used to try to systematize the policy of a hypothetical city hall with respect to cases which are typically handled through negotiations with developers, community groups and the like. This type of modeling tool can also be used for the purposes of typological research. Given a set of regulations captured in the form of rules in the model, the system can be applied to different types of contexts in order to study the different urban fabrics which are produced as a result. As an exercise, the regulations developed for one city can be adapted and applied to another, revealing typological and normative invariants. In an analogous manner, the development of a given urban context can be observed under two or more different sets of regulations in order to form an understanding of the impact of some kinds of norms, and the relative adaptability of others. Studying the concept of simulation and modeling can lead to an exploration of the notion of urban regulation itself, leading to a deeper understanding of planning and urban design. Modeling systems such as DeaRB can thus simultaneously serve as tools for exploring urban design, and be themselves the subjects of analysis and design studies.

References

Ableson, Harold and Jerald J. Sussman, Structure and Interpretation of Computer Programs, MIT Press, Cambridge, MA, 1985.

Boston Redevelopment Authority, Downtown Zoning Midtown Cultural District Plan, 1989.

Fargas, Josep Maria, "Designing Rules: Heuristics of Invention in Design", Masters thesis, Department of Urban Studies and Planning, MIT, Cambridge, MA, 1991.

Fargas, Josep Maria and Pegor Papazian, "Metaphors in Design: An Experiment with a Frame, Two Lines and Two Rectangles", ACADIA'92, Charleston, SC, 1992.

Kwartler, Michael, Daylight as a Zoning Device for Midtown, Kwartler Associates and Rensselaer Polytechnic, New York, NY, 1988.

Lai, Richard Tseng-yu, Law in Urban Design and Planning: The Invisible Web, Van Nostrand Reinhold Company Inc., London, England, 1988.

Lynch, Kevin and Gary Hack, Site Planning, MIT Press, Cambridge, MA, 1985.

Order a complete set of
eCAADe Proceedings (1983 - 2000)
on CD-Rom!

Further information:
http://www.ecaade.org