Clash Detection and Code Checking BIM Platform for the Italian Market

Caterina Trebbi, Michelangelo Cianciulli, Francesco Matarazzo, Claudio Mirarchi, Guido Cianciulli and Alberto Pavan

Abstract Development of experimental products for the BIM environment, and plug-ins are integrated with a collaborative platform to create, manage and validate BIM models in a common data environment.

Keywords ACDat · CDE · Clash detection · Code checking · IFC · IDM · MVD · Rule checking · Model checking · mvdXML · Bsi · BuildingSMART · BCF

1 Introduction

The term BIM[1] means a process of creating and managing a digital model containing all the information relating to a construction. It is essential to create a single environment in which it is possible to manage all data and information centrally throughout the life cycle of a building. The UK voluntary standard BS 1192-1:2007 (BS 2007) originally theorized this environment as Common Data Environment (CDE). The Italian voluntary technical regulation UNI 11337 introduces this environment in the process of digitization of the construction sector as *Ambiente di Condivisione Dati* (ACDat), outlining its characteristics, objectives and advantages. An *IT infrastructure for data collection and management* (UNI 11337-5:2017), where all those accredited can share information, in accordance with specified procedures. The platform ensures transparency and a reduction in errors. All actors involved in the process can relate, sharing data and design, construction or maintenance choices. In a BIM process, it is also essential to coordinate data and information contained in different

[1] Building Information Modeling

C. Trebbi (✉) · C. Mirarchi · A. Pavan
Architecture, Built Environment and Construction Engineering—ABC Department, Politecnico di Milano, Milano, Italy
e-mail: caterina.trebbi@polimi.it

M. Cianciulli · F. Matarazzo · G. Cianciulli
ACCA Software, Bagnoli Irpino, Italy

B. Daniotti et al. (eds.), *Digital Transformation of the Design, Construction and Management Processes of the Built Environment*, Research for Development,
https://doi.org/10.1007/978-3-030-33570-0_11

BIM models: coordination in the model, between the models and towards the relevant regulations.

Model checking is now an integral part and key element in information modeling and management (Ciribini et al. 2014). In standard design processes, only 5–10% of the project information content is systematically checked (BuildingSmart Finland 2012). The model checking, however, allows an automatic validation of 40–60% of the project (IBIDEM). Part 5 of UNI 11337 defines these activities as *Clash detection* and *Code checking* and plans to carry out these activities within the *Coordination Levels* (UNI 11337-4:2017, UNI 11337-5:2017).

Clash detection means the verification of geometric interferences and can be a coordination problem in 3D BIM (Akponeware and Zulfikar 2017). Two objects can collide because they occupy the same physical space (*hard clash*): a pipe that passes through a beam. On the other hand, two objects generate interference because they do not collide, but they are too close together (*clearance clash* or *soft clash*): space for maintenance of a plant element is not enough. Conflicts can also develop over time (4D or workflow clash): two objects designed to overlap during assembly.

Code checking is the verification of the compliance of the digital model with the corresponding regulation. Using specific software to make these checks allows you to speed up times and limit errors, improving the efficiency of building design (Greenwood et al. 2010). Automated rule checking has been identified as potentially providing significant value to the AEC (Solihin and Eastman 2015). It is essential to verify the conformity of the models with regulatory or technical requirements. The kitchen must have a minimum useful area of five square meters (building code of the City of Milan). It may be necessary to verify compliance with the requirements of *Employer Information Requirements* (*E.I.R.*) or of *BIM Execution Plan* (*B.E.P.*) (PAS 1192-2:2013). Particular modeling criteria may be required. It may be useful to verify the correspondence with custom requirements defined, for example by the client in the contract documents. The market does not appreciate balconies with a depth less than 1.40 m.

It is essential to identify such interferences and inconsistencies, to share them with the specialists concerned and, if necessary, to proceed with their solution in a coordinated manner. This way the potential flaws are detected in advance and a reliable performance in the following applications is guaranteed through an information model achieving fully coordinated information management (Ciribini et al. 2014). *usBIM.platform* is an ACDat cloud that integrates with several plug-ins for the management of the process that extends from the design to the realization and subsequent management of the item. Specifically, the Politecnico di Milano is providing the development of *Code checking* (*usBIM.code*) and *Clash detection* (*usBIM.clash*) plug-ins.

2 Methods

We have studied the software and information systems currently on the market (Table 1).

These are, for most of their activities, highly evolved documentary archives. Most of the computer solutions proposed are overseas and are related to software houses that produce graphic modeling systems.

Based on the guidelines of national (UNI 11337-4, 5:2017) and international (PAS 1192) regulations, we have carried out various hypotheses of structuring the CDE and WORKFLOW for the verification of the design documentation (Fig. 1).

After the analysis phase, we tested the platform by setting up several case studies. We have assumed and structured the working folders, the associated permissions, set the workflows for the definition of protocols and procedures for the approval of documents through a GATE construction, special folders with a GATE area where documents pending for approval are placed. In addition, we have developed a file encoding system with the platform's #TagBIM system. Thanks to the tests we worked out how to integrate, on this platform, additional model checking systems in order to completely manage the BIM process. Furthermore, the tests conducted demonstrated the possibility of implementing a number of useful functions for coordination, such as the implementation of *Markups* and *Issues* (visible annotations on the shared model to a specific user, containing information for the management of the problem). We have shared the implementation proposals with the software house.

We have carried out an accurate study of the administrative rules and procedures in order to determine the parameters necessary for code checking and the telematics validation of the digital model of the building. In the first phase we focused on Italian regulations (Milan, Naples and Rome), and then on international regulations: Berlin, Madrid, London and Paris. For each country, we looked for the guidance regulations relating to each discipline (building, hygiene, urban planning, architectural barriers and fire prevention). Once we had defined and analyzed the standards, we selected the articles that could be computable by the machine. Articles were defined by verifiable parameters without needing the aid of a human analysis (such as an

Table 1 Analyzed software systems

BIM360 (*Autodesk*)	BIMx (*NEMETSCHECK*)	TRIMBLE CONNECT (*Trimble*)
ProjectWise (*Bentley*)	Aconex (*Oracle Aconex*)	usBIM.platform (*ACCA software*)
BIMPLUS (*ALLPLAN*)	STR Vision Teamwork (str)	

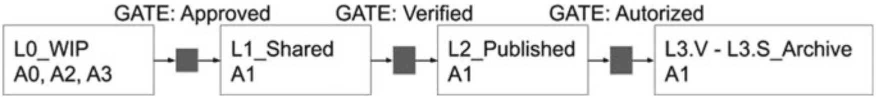


Fig. 1 Structuring hypothesis

analysis conducted by a municipal technician). Each of them was "disarticulated" into a list. It contains the subject, the number and the full text of the prescription; the measurable parameter quoted in the prescription (surface, volume, etc.) and its unit of measurement. Moreover, in the last two columns, the indication for the respect of the article is identified along with the definition of the parameter under consideration (Table 2).

We have made a comparison between different cities in order to determine code checking rules that are valid for different countries. This analysis showed that, for different building regulations, there are many similarities but also many differences.

Table 2 Table of disarticulations: building code of the City of Milan and D. 14 June 1989 on the elimination of architectural barriers

Min. Room Size	97—The usable area […] bedroom for 1 person: 8 mq;	Surf. (m^2)	$S_{u1p} \geq 8$ mq	$\mathbf{S_{u1p}}$: Usable area – bedroom for 1 person; **Usable floor area** = floor area without surrounding walls
Inatural and Direct Lighting	105.1—[…] 1/10 of the floor surface […] The distance between the window and the furthest point […] 2.5 the height from the ground of the highest point of the window surface	Surf. (m^2)	S_{wtot}-S_c-(2/3)$S_a \geq$ 1/10 $\frac{1}{10}$ S_{loc} $D_{loc} \leq 2.5\ H_w$	$\mathbf{S_{wtot}}$: Total window area $\mathbf{S_c}$: Window surface less than 0.60 m from the floor level $\mathbf{S_a}$: Window area calculated by Art.105.4 $\mathbf{S_{loc}}$: Floor surface $\mathbf{D_{loc}}$: Local depth $\mathbf{H_w}$: Highest point of the window surface
Bathrooms	8.1.6—[…] The axle of the W.C. cup or bidet must be placed at a minimum distance of 40 cm from the lateral wall; The front edge at 75–80 cm from the rear wall and the top floor at 45–50 cm from the floor	Length (m)	$D_{l\text{-}bidet}$, $D_{l\text{-}wc} \geq$ 0.40 $0.75 \leq D_{a\text{-}bidet}$, $D_{a\text{-}wc} \leq 0.85$ $0.45 \leq D_{s\text{-}wc}$, $D_{s\text{-}bidet} \leq 0.50$	$\mathbf{D_{l\text{-}bidet}}$, $\mathbf{D_{l\text{-}wc}}$: Distance between side wall and bidet/WC cup axis $\mathbf{D_{a\text{-}bidet}}$, $\mathbf{D_{a\text{-}wc}}$: Distance between front edge of bidet/WC and rear wall $\mathbf{D_{s\text{-}bidet}}$, $\mathbf{D_{s\text{-}wc}}$: Distance between bidet/toilet top and floor

This is a fundamental aspect because all of this could result in the creation of a different information delivery manual (ISO 29481-1:2016; Berard and Karlshoej 2012; Karlshoej 2011) and a different model view definition (BuildingSMART international) with the aim of code checking models for different building regulations. The idea was therefore to develop BIM tools that allow, starting from a generic IFC model now produced by all BIM authoring software, to enrich it, to make it conform to another view (MVD) as required by regulatory organizations. In this way, the regulatory organization itself is interested in investing in the development of such BIM tools, as well as the software houses concerned. Anyone can make a significant contribution at any time to the development of these tools because an open format is used.

The developed system is *usBIM.code*. It consists of two separate modules that perform different functions: *CODEmaker* and *CODEcontroller*.

CODEmaker is the BIM tool that allows you to digitize, in IFC, the missing re-requested information. The system works on an IFC file generated by any BIM authoring software; it enriches it with the addition of graphical objects and information needed for *Code checking*; it saves an IFC containing all the additional data. The objects and information needed for the checks are specific to the various building regulations, and has emerged from the previous research phase. In this way, any software can use the new information to perform code checking. *CODEcontroller* is the BIM tool that uses the input information for code checking and related analysis. We have paid particular attention to the issue of the inclusion of information in the IFC file, which is an important issue discussed with the software house. As the analysis showed, building regulations often do not explain clearly which surfaces should be included in the calculation of aero-illuminating surfaces. If a window presents an opaque section, it is cautious to exclude it from the calculation of the illuminating surface. Allowing the user the possibility to define precisely the area of the window that contributes to the lighting allows the system to make a more precise and cautious calculation.

Once we had received the beta version of the *usBIM.code* program we performed tests. We have developed models of both residential and mixed-use buildings (offices, shops and residences), using both Revit (Autodesk) and Edificius (ACCA software) to test their compatibility. After completing the tests, we discussed the results with the software company for the necessary improvements (Fig. 2).

The ACCA software system for interference verification is *usBIM.clash*. The plug-in can detect any *hard clash*, *hard clash* with tolerance and *clearance clash* present on the IFC model (or between models). In order to perform its verification the program executes computational geometry algorithms considering all the elements one by one that are part of the selection made. Using specific data structures, the system processes a list for each element considered. The list contains the remaining elements of the model that could give rise to a clash. This analysis, quick but approximate, considers the selected verification settings. For example, in order to verify whether two objects collide, before making a more precise but also computationally more onerous verification, the respective bounding boxes are calculated and it is verified whether these are not disjointed. In order to achieve a better level of performance, the

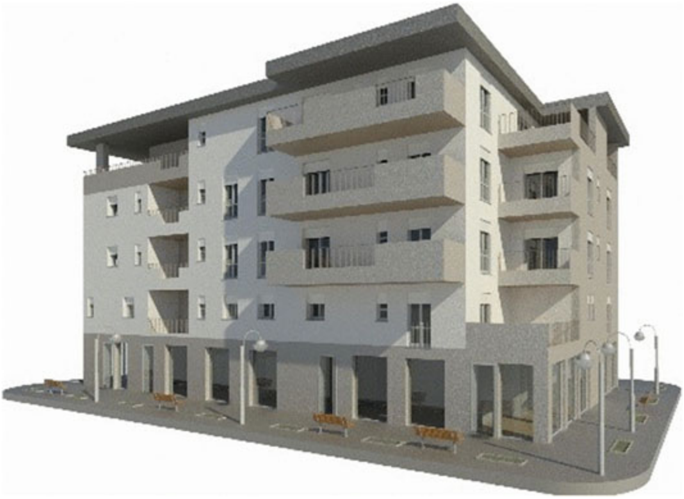
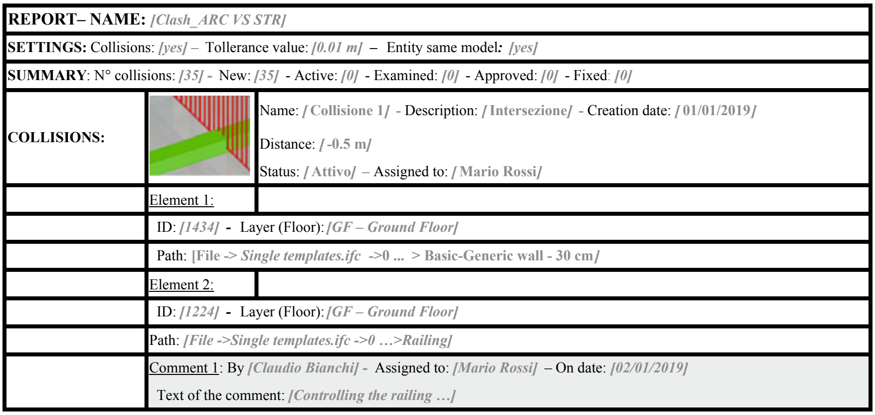


Fig. 2 Test model (IFC2X3)

algorithms make considerations of this kind over several phases, for all three types of interference (hard clash, hard clash with tolerance and clearance clash). Then, after the system has populated the list with the elements that at a rough analysis generate interference with the current element, we proceed with progressively more precise and increasingly expensive assessments. When a check fails it means that the two elements considered do not generate interference and therefore you can delete the element from the list and continue repeating it with the next. If you repeat the process for all the elements in the list and the list is not empty, there is some interference. In the algorithms used, attention has been paid to the management of calculation errors due to the finished precision of computers when dealing with decimal numbers.

Once the beta program arrived from the software house, we conducted tests on simplified models, specifically modeled in both Revit and Edificius in order to verify and perfect the three different verification algorithms. We first verified the interference on entities of the same model, then on entities of different models. We performed clash detection with both *usBIM.clash* and *Navisworks Manage 2018*, in order to compare the results (Fig. 3).

We simulated the verification activities required by UNI 11337 in the coordination levels of a BIM process. Tests have shown the need to implement a number of features to better manage the resolution of the interference. For example, this includes the possibility of assigning an interference to a specific user by adding notes or creating interference groups (e.g. according to the user who must solve them or the elements that generate it). To make communication between process actors more efficient, we have suggested a standard report for collision transmission, containing all the useful data which can be exported. The user can set the content of the report by filtering the information they consider to be useful.



REPORT– NAME: *[Clash_ARC VS STR]*		
SETTINGS: Collisions: *[yes]* – Tollerance value: *[0.01 m]* – Entity same model**:** *[yes]*		
SUMMARY: N° collisions: *[35]* - New: *[35]* - Active: *[0]* - Examined: *[0]* - Approved: *[0]* - Fixed: *[0]*		
COLLISIONS:		Name: **/ Collisione 1/** - Description: **/ Intersezione/** - Creation date: **/ 01/01/2019/** Distance: **/ -0.5 m/** Status: **/ Attivo/** – Assigned to: **/ Mario Rossi/**
	Element 1:	
	ID: *[1434]* - Layer (Floor): *[GF – Ground Floor]*	
	Path: **[File ->** ***Single templates.ifc*** **->0 ... > Basic-Generic wall - 30 cm/**	
	Element 2:	
	ID: *[1224]* - Layer (Floor): *[GF – Ground Floor]*	
	Path: *[File ->Single templates.ifc ->0 …>Railing]*	
	Comment 1: By *[Claudio Bianchi]* - Assigned to: *[Mario Rossi]* – On date: *[02/01/2019]* Text of the comment: *[Controlling the railing …]*	

Fig. 3 Standard interference verification report

3 Results

The *usBIM.platform* system allows you to customize the data sharing environment according to your project needs. The use of *GATE* folders, in combination with the definition of associated *permissions* and the definition of *workflows*, allows for a correct application of procedures/processes of revision and validation of documentation. Unlike most commercially available CDEs, procedures can be freely structured within the platform according to the needs of the individual project. The system supports open formats (BCF, IFC, etc.). Unlike the main solutions proposed by the market, it is not only a "container of information"; but a management system for the informative content of a project throughout its life cycle. The IFC project model can be displayed in the cloud without having BIM authoring software and can be federated with other project models.

Being able to perform model checking tests in the sharing platform leads to a new way of working based on sharing information and transparency of processes. Creating standard sets of rules according to the reference standard contributes to the optimization of verification processes. Most of the solutions in commerce today are foreign, concentrated on an international market, far from national legislative requirements. *usBIM.code*, currently under development, verifies the correspondence of projects to local legislation and/or specific design needs quickly, without compromising the quality of assessment. You can choose the technical or legislative rule to verify and import the IFC model file into the software. The system will analyze and highlight verification data and calculate missing data. If necessary, the user will enter the missing data into a graphical form. In the section labeled "diagnostic" it will be possible to visualize a list of the requirements that are not respected and the objects (or parts of the project) that are not compliant. Checks can also be carried out on models developed by BIM authoring software with insufficient information. The insertion of missing data is done on an open format (IFC) to ensure the process is intelligible.

The problems associated with the exchange of information, currently investigated (Belsky et al. 2016; Bloch and Sacks 2018; Sacks et al. 2017), are thus overcome. The implemented information also remains stored in a new IFC file which can also be read by other software.

The actors taking part in the process are varied; each of them elaborates its own BIM model using a specific software of BIM authoring according to its needs. It should be stressed that in a real process not all detected interferences should be considered in the same way. Some, for example, are "false positives" because they are easily manageable on the construction site. Other collisions, on the other hand, should not be neglected and should be resolved at the design phase. *usBIM.clash* works with models in IFC format; it can verify interferences both between entities belonging to the same model and between entities belonging to different digital models. It is a simple and immediate system, unlike most clash detection software. IFC models for which verification is to be conducted are imported into the software; through the appropriate commands, the criteria for selecting the entities on which to process conflict verifications are defined. These criteria may be recorded and used subsequently to repeat the verifications on the updated models. The type of clash to be identified must be defined: *hard clash*, *hard clash* with tolerance or *clearance clash*. The system then produces a table with all interferences. By selecting the interference, the system provides a view of the model entities involved in the collision. In addition, each interference reports a set of information useful for its management. You can export an interference report in different formats (pdf, html, xml, csv, xlsx). In the future, the software will also be able to export in BCF standard form.

4 Conclusion

BIM not only results in three-dimensional modeling, but also, and above all, in effective information management, in collaboration and coordination between professionals, in process automation. With this study we have looked into new collaborative procedural practices in order to understand how to better exploit this new way of working. The presence on the CDE of all information relating to the intervention, being complete and up to date, allows for an informed management of the whole process. Every activity must be traced allowing for maximum control over operations; the data must be organized and catalogued clearly; the roles and work processes of those involved in the process must be defined; the platform structure must be adapted to requirements (B.E.P., E.I.R., etc.) or internal needs.

We theorized a system where, in a single environment, in addition to sharing information it is possible to use model checking tools. In a traditional design process, process assessments are carried out manually and on a sample basis. This makes the process slow and fertile to the proliferation of errors. The tests we have carried out confirm that delegating massive project assessments to a machine allows users to optimize time and costs. The study, through the analysis of rules and administrative procedures, has allowed us to define the parameters necessary for the automatic validation of digital models. The goal is to create software capable of adapting to different needs; to create a proven set of rules specific to different regulations or needs.

The analyses and the work carried out confirm that the path is still long but the goal is attainable. The advantages that can be obtained are numerous, starting from the design phase, all the way up to the realization phase and, consequently, also to the management and maintenance phases.

References

Akponeware AO, Zulfikar AA (2017) Clash detection or clash avoidance? an investigation into coordination problems in 3D BIM. Buildings 7(4):75. https://doi.org/10.3390/buildings7030075

Belsky M, Sacks R, Brilakis I (2016) Semantic enrichment for building information modeling. Computer-Aided Civil Infrastruct Eng 31:261–274. https://doi.org/10.1111/mice.12128

Bloch T, Sacks R (2018) Comparing machine learning and rule-based inferencing for semantic enrichment of BIM models. Autom in Constr 91:256–272. https://doi.org/10.1016/j.autcon.2018.03.018

Berard O, Karlshoej J (2012) Information delivery manuals to integrate building product information into design. Electron J Inform Technol Constr 17:64–74

BS 1192:2007, Collaborative production of architectural, engineering and construction information – Code of practice

BuildingSMART International, Model View Definition (MVD)—An Introduction. https://technical.buildingsmart.org/standards/mvd/

Finland BuildingSmart (2012) COBIM series 06—quality assurance. Common BIM Requir 1:1–27. https://doi.org/10.1073/pnas.0703993104

Ciribini ALC, Ventura SMB, Marzia B (2014) Informative content validation is the key to success in a BIM-based project. Territorio Italia 1:87–111. https://doi.org/10.14609/Ti_2_15_1e

de Farias TM, Roxin A, Nicolle C (2018) A rule-based methodology to extract building model views. Autom Constr 92:214–229. https://doi.org/10.1016/j.autcon.2018.03.035

Greenwood D, Lockley S, Malsane S, Matthews J (2010) Automated compliance checking using building information models. In: Construction, building and real estate research conference

Khemlani L (2015) Automating code compliance in AEC, AECbytes Feature

Karlshoej J (2011) Information delivery manuals. http://iug.buildingsmart.org/idms

Nawari N (2012) The challenge of computerizing building codes in a BIM environment. American Society of Civil Engineers (ASCE), pp 285–292. https://doi.org/10.1061/9780784412343.0036

Sacks R, Ma L, Yosef R, Borrmann A, Daum S, Kattel U (2017) Semantic enrichment for building information modeling: procedure for compiling inference rules and operators for complex geometry. J Comput Civil Eng 31:04017062. https://doi.org/10.1061/(asce)cp.1943-5487.0000705

Solihin W, Eastman C (2015) Classification of rules for automated BIM rule checking development. Autom in Constr 53:69–82. https://doi.org/10.1016/j.autcon.2015.03.003

Zhang S, Teizer J, Lee JK, Eastman CM, Venugopal M (2013) Building Information Modeling (BIM) and safety: automatic safety checking of construction models and schedules. Autom Constr 29:183–195. https://doi.org/10.1016/j.autcon.2012.05.006

Standards and Laws

UNI 11337:2017-4 - Building and civil engineering works – Digital management of the informative processes - Part 4: Evolution and development of information within models, document and objects. UNI - Ente Italiano di Normazione

UNI 11337:2017-5 -Part 5: Informative flows in the digital processes

PAS 1192-2:2013 - Specification for Information Management for the capital/delivery phase of construction projects using Building Information Modelling

ISO 29481-1:2016 - Information delivery manual – Part 1: Methodology and format

Italy:

D.Lgs 18/04/2016, 50 - Codice dei contratti pubblici

D.M. 02/04/1968, 1444 - Limiti inderogabili di densità edilizia, di altezza, di distanza fra i fabbricati e rapporti massimi tra gli spazi destinati agli insediamenti residenziali e produttivi e spazi pubblici o riservati alle attività collettive, al verde pubblico o a parcheggi, da osservare ai fini della formazione dei nuovi strumenti urbanistici o della revisione di quelli esistenti

G.U. 16/11/ 2016, S. G. 268 - Quadro delle definizioni uniformi

Codice Civile Italiano

L. 17/08/1942, 1150 - Legge urbanistica

L. 24/03/1989, 122 - Disposizioni in materia di parcheggi, programma triennale per le aree urbane maggiormente popolate, nonché modificazioni di alcune norme del testo unico sulla disciplina della circolazione stradale

D.M. 16/05/1987, 246 - Norme di sicurezza antincendi per gli edifici di civile abitazione

D..M. 14/06/1989, 236 - Prescrizioni tecniche necessarie a garantire l'accessibilità, l'adattabilità e la visitabilità degli edifici privati e di edilizia residenziale pubblica, ai fini del superamento e dell'eliminazione delle barriere architettoniche

D.M. 05/07/1975 - Modificazioni alle istruzioni ministeriali 20/06/1896, relativamente all'altezza minima ed ai requisiti igienico-sanitari principali dei locali di abitazione

Milan:

Piano di Governo di Territorio - Piano delle Regole - Norme Tecniche di Attuazione del Comune di Milano

Regolamento Edilizio del Comune di Milano

Regolamento Locale di Igiene della Regione Lombardia

L.R. 20/02/1989, 6 - Norme sull'eliminazione delle barriere architettoniche e prescrizioni tecniche di attuazione

Rome:

Piano Regolatore Generale - Norme Tecniche di Attuazione del Comune di Roma

Regolamento Generale Edilizio del Comune di Roma

Naples:

Norme Tecniche di Attuazione del Comune di Napoli

Regolamento Edilizio del Comune di Napoli

L.R. 20/03/1982, 14—Indirizzi programmatici e direttive fondamentali relative all' esercizio delle funzioni delegate in materia di urbanistica