

QUALITY ASSURANCE E MODEL CHECKING

Nell'approccio tradizionale alla progettazione la verifica di conformità del progetto è condotta in modo manuale e basata sulla rappresentazione grafica 2D.

Questo tipo di procedura non automatizzata può comportare l'inefficienza delle attività in termini di:

- ambiguità;
- soggettività nelle interpretazioni;
- aumento di tempi;
- aumento di costi.

Lo sviluppo della metodologia BIM ha contribuito a diffondere l'interesse verso

tecnologie per
l'analisi semi-automatizzata condotta attraverso strumenti di

Model Checking

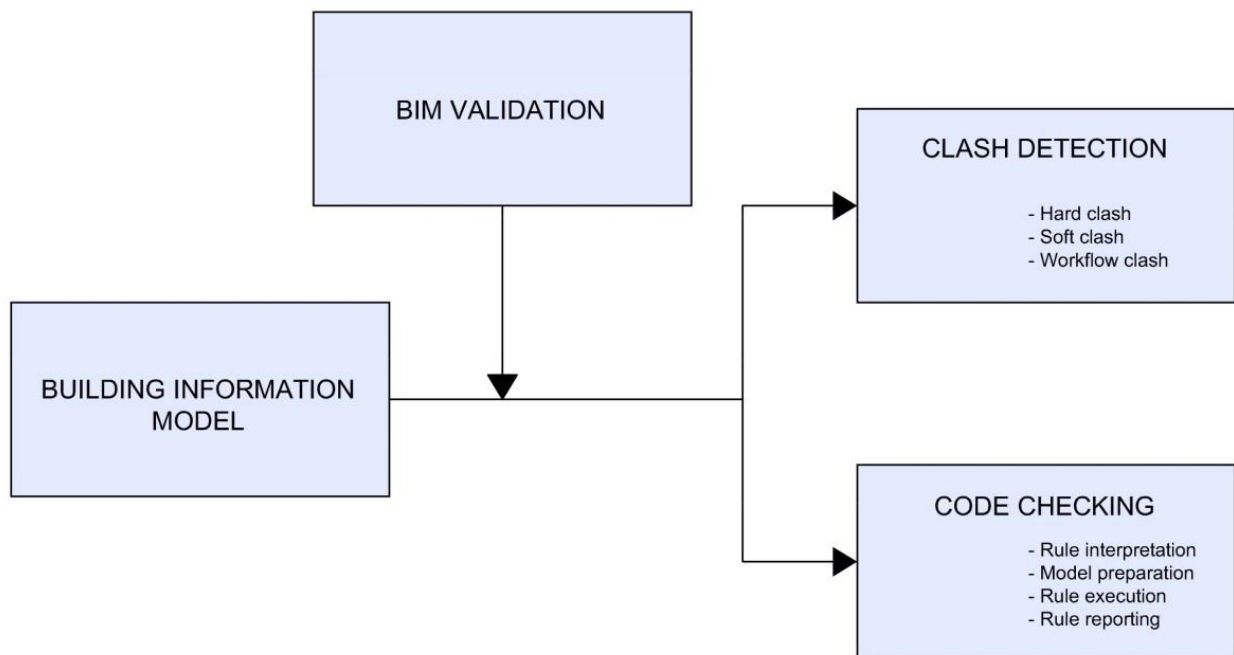
comportando lo sviluppo di nuovi software per

Quality Assurance e Quality Control (QA/QC)

basate su
regole parametriche (Rule-based).

I software dedicati a questo tipo di valutazioni sono utilizzati in maniera diffusa per:

- **BIM Validation**
verifica del livello di coerenza interna del modello;
- **Clash Detection**
controllo delle interferenze finalizzato all'individuazione delle intersezioni fisiche e di potenziali conflitti nella progettazione di modelli relativi a ciascuna disciplina e nell'integrazione di modelli derivanti da differenti discipline;
- **Code Checking**
verifica di conformità del progetto, in termini di modello, alle normative di riferimento e al quadro esigenziale della committenza.



Queste attività consentono di avere un modello **“affidabile”** per proseguire poi con le elaborazioni e le analisi relative alle successive dimensioni del BIM: 4D, 5D, 6D e 7D.

BIM Validation

Questo processo di Model Checking per la Quality Assurance (QA) determina un

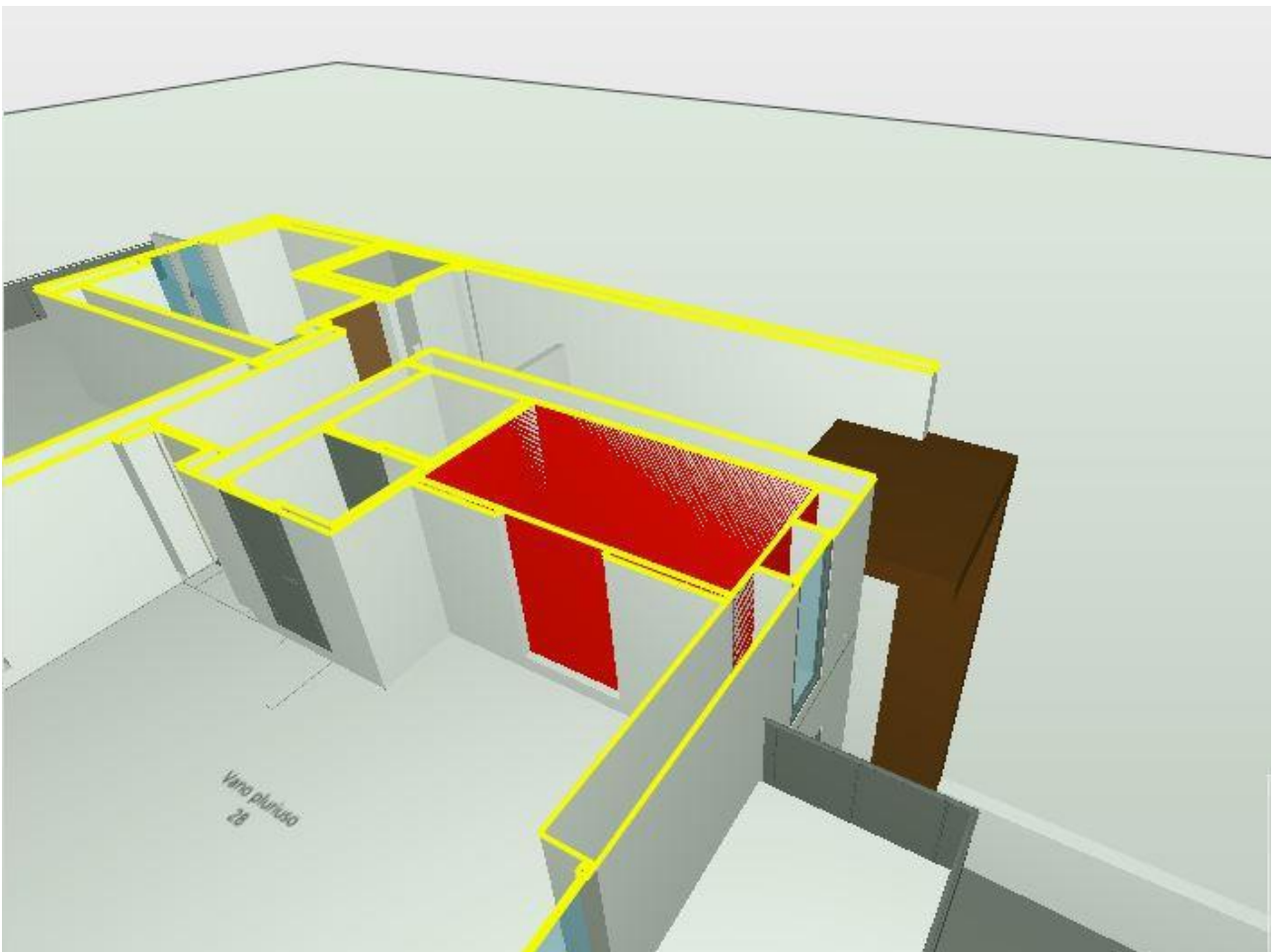
pre-check del contenuto informativo del modello,
attraverso il set di **regole** di BIM Validation,

finalizzato

a garantire la produzione di un Building Information Model affidabile per le successive analisi (*Quantity Take Off, Building Energy Modeling, ecc.*) e check avanzati (*Code Checking*).

Solo in seguito a questa validazione sarà possibile effettuare i controlli specifici del
Code Checking.

È una fase essenziale che deve essere svolta prima della fase di Code Checking in quanto verifica il livello di coerenza e qualità interna, attraverso **l'applicazione di rule-set BIM Validation**, personalizzabile dall'utente e applicabile alle diverse discipline (architettonica, strutturale, impiantistica).
Alcuni software di QA contengono rule-set predefiniti necessari alla BIM Validation.



BIM Validation: Assenza di una porta nel vano evidenziato (software Solibri)

Le criticità riscontrabili all'interno del modello possono riguardare:

- Aspetti alfanumerici (che rientrano nei BIM requirement individuati in precedenza);
- Aspetti connessi alla geometria riscontrando errori di:
 - Modellazione (utilizzo non corretto del software di modellazione – es. due muri sovrapposti),
 - Errori di progettazione nel processo di BIM Validation possono essere individuati all'interno della stessa disciplina o all'interno di discipline differenti (es. la non coerenza tra l'altezza di un muro e la quota del solaio).

Inoltre, la BIM Validation consente di analizzare **la completezza del contenuto informativo** associato ad un oggetto parametrico e quindi, attraverso opportuni sistemi di classificazione, permette anche di validarne il relativo

Level Of Development (LOD).

Si considera, ad esempio, un componente “porta” per il quale ad un determinato LOD corrispondono degli attributi quali “Resistenza al fuoco”, “Uscita antincendio”, “Operazioni legate al funzionamento della porta”:

la validazione del contenuto informativo verifica l'effettiva presenza e corretta compilazione di tali parametri al fine di un confronto tra quanto dichiarato e quanto effettivamente modellato, ponendosi a supporto di un corretto flusso informativo tra le parti interessate.

BIM Validation - Architectural	Acc	Rej	Maj	Nor	Min	Comment
Model Structure Check					x	
Model Hierarchy	OK					
Building Floors	OK					
Doors and Windows	OK					
Door Opening Direction Definition					x	
Unique GUID values	OK					
Amount of Site Instances	OK					
Amount of Doors or Windows in Openings	OK					
If Decomposed Object has Geometry Defined, Its Parts Should Not Have Geometry	-					
If Parts of Decomposed Object have Geometry Defined, the Decomposed Object Itself Should Not Have Geometry	OK					
Material of Decomposed Objects Should Only Be defined in Part Level	OK					
Openings in Complex Walls Shouls be Related to Wall, Not Parts	-					
Component Check				x	x	
Component Dimensions				x	x	
Wall Dimensions Should Be Sensible				x	x	
Wall Height					x	
Wall Thickness	OK					
Wall Length	OK					
Wall Opening Distances				x		
Door And Window Openings Must Have at Least Minimal Size	OK					
Window Width	OK					
Window Height	OK					
Door Width	OK					
Door Height	OK					
Slab Dimensions Should Be Sensible				x		

Esempio di report finale di BIM Validation (software Solibri)

Clash Detection

Il controllo delle interferenze, quindi
la Clash Detection,

rappresenta uno degli usi più diffusi del BIM ed è

finalizzato

all'analisi di coerenza interna spaziale e geometrica del modello.

La verifica delle interferenze, assai utile nella progettazione di un modello per singola disciplina, è, in particolare, motivata allorchè coesistano insieme differenti modelli, derivanti da diverse discipline, che devono poi essere convogliati in un unico modello completo (ma, come già ricordato, applichiamo la clash detection anche a singoli modelli riferiti ad una sola disciplina – ad es. architettonico).

I clashes individuati, se non risolti, potrebbero gravemente influenzare il processo di costruzione, causare ritardi, richiedendo **modifiche al progetto in corso d'opera** con conseguente inefficienza (equivoci, errori, aumento dei costi e dei tempi).

La Clash Detection **anticipa e risolve** in maniera puntuale e semi-automatizzata ciò che in un processo tradizionale è effettuato manualmente e a campione, generalmente in una fase del processo non flessibile.

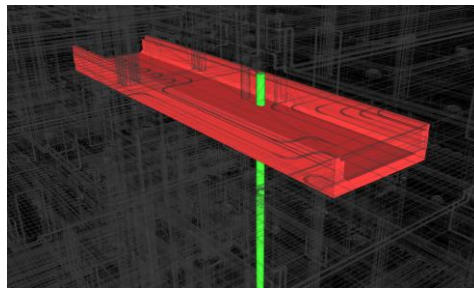
Prima di effettuare la Clash Detection sul merged model si procede alla validazione dei singoli modelli disciplinari. Infatti, un'analisi delle interferenze avanzata necessita di modelli, che saranno successivamente integrati, accurati dal punto di vista geometrico che già al loro interno dispongano del massimo grado di coerenza e senza intersezioni.

Si può parlare di **Advanced Clash Detection** quando lo strumento di Model Checking è in grado di rilevare e distinguere diversamente le interferenze, le quali possono essere suddivise in tre classi:

- **Hard Clash**

Effettiva intersezione di due o più oggetti, appartenenti alla stessa o a diverse discipline (occupazione dello stesso spazio da parte di due o più oggetti).

Esempi di hard clash possono essere rappresentati, ad esempio, da una tubazione che si sovrappone ad un condotto dell'areazione o l'impianto antincendio che entra in conflitto con le strutture edilizie.



- **Soft Clash\Clearance Clash**

Interferenze caratterizzate da una vicinanza inadeguata.

L'oggetto necessita di una *buffer zone* per il suo corretto funzionamento, per l'accessibilità, per l'isolamento, per la manutenzione o per la sicurezza.

- **4D\Workflow Clash**

Interferenze che si verificano tra le varie fasi delle lavorazioni al momento della costruzione.

È una interferenza spesso sottovalutata ma, solitamente, è anche la più costosa e difficile da risolvere.

-Si potrebbe essere **impossibilitati** nel montare un determinato componente se prima ne viene montato un altro,

-la consegna dei materiali potrebbe essere **programmata in ritardo** rispetto alla lavorazione di riferimento,

-o potrebbe **essere molto pericoloso** per gli operai, in termini di sicurezza, effettuare una lavorazione in prossimità di macchinari in movimento (interferenza spaziale o interferenza temporale in cantiere).

Vengono effettuati **controlli ciclici del modello** atti a evidenziare eventuali interferenze inter-disciplinari, ponendosi alla base di riunioni di coordinamento durante le quali tutte le parti coinvolte propongono delle soluzioni.

Una volta che le soluzioni sono state testate nell'ambiente virtuale, possono essere implementate e successivamente verificate.

Code Checking

Il Code Checking,
come declinazione del Model Checking, è
finalizzato alla

**validazione della progettazione attraverso la comparazione dei
parametri contenuti all'interno del modello con**

**-normative,
-codici di riferimento
-e requisiti della committenza.**

Le informazioni,
contenute nei documenti normativi,

e

tradotte in regole parametriche mediante un sistema semantico di lettura,

consentono, attraverso un controllo basato su regole, di eseguire un controllo (check) con la possibilità di estrarre automaticamente i risultati sotto-forma di report.

I risultati di un check possono essere: *pass*, *fail*, *warning* e *unknown* (in caso di dati mancanti).

Il processo Rule-based Code Checking si compone di quattro fasi:

- **Fase 1 - Rule interpretation**
la fase di interpretazione consiste nel strutturare le regole in maniera logica per la loro implementazione nel software di Rule Checking;
- **Fase 2 - Model preparation**
il modello è preparato in termini di presenza e integrazione di tutti gli attributi necessari all'esecuzione del check;
- **Fase 3 - Rule execution**
viene eseguito il check applicando la regola;
- **Fase 4 - Rule reporting**
i risultati derivanti dal controllo vengono raccolti sotto forma di report per essere condivisi eventualmente con gli altri membri del team.

Fase 1 - Rule interpretation

Consiste nella
trasposizione del testo normativo in regole parametriche

e successiva
implementazione delle stesse nel software.

Le informazioni necessarie sono estrapolate dal testo normativo attraverso una analisi logica e sintattica. Le regole, una volta dedotte da affermazioni logiche e trasformate in parametriche, sono applicate al contenuto semantico del modello BIM.





Una delle metodologie di analisi logico-semantica, per l'interpretazione della normativa e conseguente traduzione della stessa in regole implementabili in un software Rule-based Code checking, è la cosiddetta **RASE Methodology**.

Essa è basata su quattro operatori semantici (tag) che rappresentano le parti combinate di un check:

- **R equirements**: il requisito di un check che “deve” o “non deve” essere soddisfatto;
- **A pplicability**: è il campo di applicazione della regola;
- **S election**: possibili alternative offerte dalla prescrizione normativa;
- **E xceptions**: inverso alla “applicabilty”, rappresentano le deroghe alla prescrizione.

La metodologia RASE prevede che

al testo normativo, attraverso delle operazioni di mark-up, viene applicato un sistema di colori (quattro, uno per ogni tag), consentendone l'analisi della struttura logica.

			
Requirement {blue}	Applies {green}	Select {red}	Exception {orange}

Rule source	ICC IECC 2006 502.5 Moisture control
Rule description	All green framed green red walls, floors red and red ceilings red orange not ventilated orange to allow moisture to escape shall be provided with an blue approved vapor retarder blue having blue a permeance rating of 1 perm blue (5.7×10^{-11} kg/Pa s m ²) or less, when tested in accordance with the desiccant method using Procedure A of ASTM E 96. The vapor retarder shall be blue installed on the warm-in-winter side blue of the insulation. Exceptions: orange Buildings located in Climate Zones 1 through 3 orange as indicated in Figure 301.1 and Table 301.1. In construction where orange moisture orange or its orange freezing orange will not damage the materials. Where other approved means to avoid orange condensation orange in unventilated framed wall, floor, roof and ceiling cavities.

Mark-up operator	Mark-up color	Identification of construction object	Property of object	Logic relation	Value
apply	green	building element	construction	=	framed
select	red	wall	(existence)	=	(true)
select	red	floor	(existence)	=	(true)
select	red	ceiling	(existence)	=	(true)
except	orange	building element	ventilated	=	(true)
except	orange	site	zone	=	1
except	orange	site	zone	=	2
except	orange	site	zone	=	3
except	orange	building element	moisture proof	=	(true)
except	orange	building element	frost proof	=	(true)
except	orange	building element	condensation proof	=	(true)
require	blue	building element. vapor retarder	(existence)	=	(true)
require	blue	building element. vapor retarder	permeance	<	1
require	blue	building element. vapor retarder	location	=	warm-in-winter

Fase 2 – Model preparation

La preparazione del modello
è rappresentata da

insieme delle operazioni finalizzate ad integrarne al suo interno
tutte le informazioni necessarie alla successiva fase Rule Checking
(inclusando anche le regole definite ad esempio dal committente).

Al fine di supportare questo tipo di processo automatizzato, infatti, il modello BIM, oltre a contenere le informazioni derivanti dal processo di modellazione (informazioni

geometriche), deve essere anche arricchito con il contenuto informativo di non immediata elaborazione.

Il modello, infatti, dovrà contenere:

- Corretta classificazione per diverse discipline in modo da supportare l'attività di Clash Detection;
- Tutte le informazioni necessarie per il controllo della regola (comprese le informazioni semantiche come il nome dell'oggetto, attributi, relazioni, ecc.).

Fase 3 - Rule execution

Le regole,
dedotte e parametrizzate nelle fasi precedenti,

vengono

raggruppate in domini di validazione cui sottoporre il modello.

Quindi, nella fase di Rule execution, i domini di validazione sono applicati al modello importato nel software di Quality Assurance (QA).

Questa procedura è resa affidabile grazie all'utilizzo di Open standard (IFC).

Fase 4 - Rule reporting

È la fase conclusiva, successiva a tutte le iterazioni di cui necessita il modello per essere aggiornato e coerente con i rule-set impostati.

Il Rule check reporting
consiste nella

fase reportistica

in cui i risultati possono essere esportati in differenti formati.

Consente

la condivisione dei risultati da parte dei membri del team di lavoro.

Per ogni criticità riscontrata, il software ne associa la localizzazione consentendo una rapida individuazione dell'errore in fase di iterazione e aggiornamento del modello.

Il processo è semi-automatico:

individuati gli errori o le omissioni, in relazione ai rule-set eseguiti, essi vengono corretti manualmente, attraverso il software di modellazione BIM, mediante una o più iterazioni.

Le attività di QA e model checking

possono essere condotte già all'interno dei software di modellazione (come Revit),

ma possono svilupparsi in modo molto più accurato mediante software specifici come SOLIBRI e NAVISWORKS.

ESEMPLIFICAZIONE

Se ipotizziamo di condurre operazioni di Code Checking, un aspetto sempre proposto dai software (ma è solo uno dei tanti aspetti del code checking) riguarda la coerenza del modello con la **normativa relativa all'accessibilità** (DM 236/89).

La verifica del rispetto normativo da parte di scale

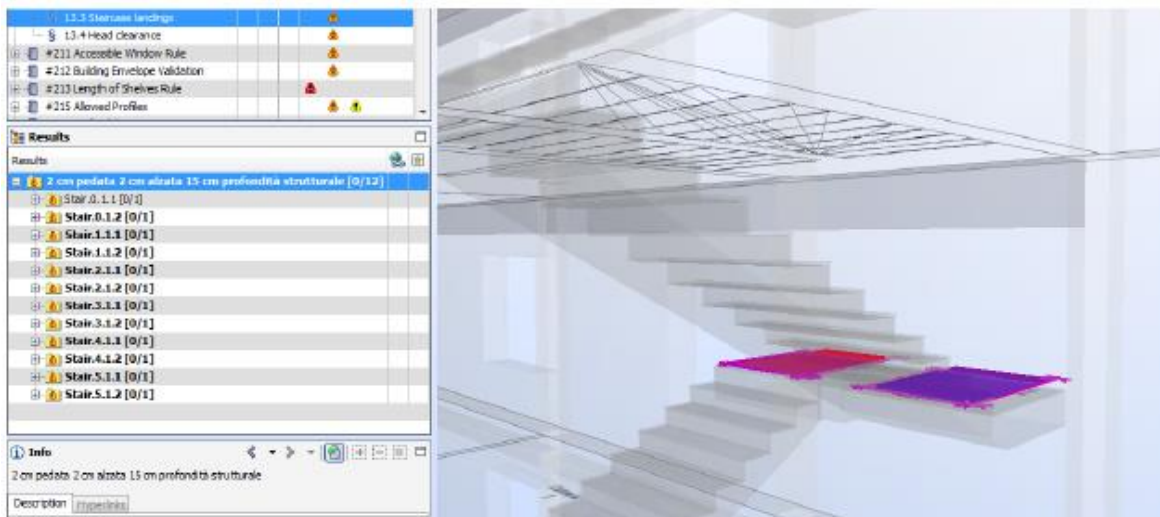
#210 Accessible Stair Rule							
§	504.2 Treads and Risers						OK
§	504.3 Open Risers						OK
§	13.1 Rise and going of steps						OK
§	13.3 Staircase landings					⚠	
§	13.4 Head clearance					⚠	

Checking rulesets "Accessible Stair Rule"

Il set di regole utilizzato per la verifica degli spazi d'uso delle componenti è composto da cinque regole:

- *Treads and Risers*: Verifica che tutte le pedate e le alzate della scala siano uniformi e che rientrino nei limiti stabiliti dalla norma;
- *Open Risers*: Verifica l'assenza di gradini aperti;
- *Rise and going steps*: Verifica una serie di parametri tra cui il rapporto alzata/pedata;
- *Staircase landings*: Verifica i parametri riguardanti il pianerottolo di riposo e la massima lunghezza della rampa;
- *Head Clearance*: Verifica che non ci siano ostacoli e sporgenze lungo il percorso della scala.

Ad esempio, se la regola *Staircase landings* rileva un sottodimensionamento del pianerottolo di riposo, con uno sviluppo in pianta pari a cm 120,00 anziché cm 150,00, la cosa ci sarà evidenziato in un report e anche graficamente.



Identificazione della regola non verificata