

Explicitando a estrutura do prédio em modelos BIM

Giving an explicit definition of a building structure in BIM models

José Luis Menegotto

Escola Politécnica - Universidade Federal do Rio de Janeiro, Brasil.

jlmenegetto@poli.ufrj.br

Abstract

This paper presents an API implementation for a BIM program written in .NET platform. The goal of this application is to launch the structure of a building automatically. The automation creates and controls the building's structural elements types using external text files with a dual purpose of being the source of information and to be the explicit description of the project. Our target is to create a definition of semantics that integrate the application with a voice user interface (VUI). The present version of the work is focused on a conventional and compact building type.

Keywords: Structural automation; Revit; Speech recognition.

Introdução

As estruturas de dados sobre as quais se sustentam os modelos tridimensionais BIM, sejam elas de padrão proprietário ou de padrão aberto e neutro como o IFC (*Industry Foundation Classes*), permitem realizar a modelagem relacional, integrada e vinculada dos componentes arquitetônicos, estruturais ou de instalações prediais. Apesar disso, pesquisas apontam que a estrutura IFC ainda carece de definições formais para que exista interoperabilidade completa entre os diversos sistemas utilizados. (Venugopal, 2012). A isto pode ser acrescentado que, no tocante à estruturação de dados que sirvam para controlar e intercambiar características morfológicas de unidades maiores da edificação, como tipologias, classes de edificações ou classes de requerimentos funcionais dos programas de necessidades, esses padrões tampouco conseguem dar uma resposta integrada. A maioria das relações paramétricas presentes nos objetos dos modelos BIM ficam limitadas ao nível individual e construtivo, podendo, somente em alguns casos, incorporar relações a sistemas maiores. A tarefa de criar estruturas de dados relacionais cada vez mais abrangentes, compete em grande parte aos agentes de projeto. Na literatura encontramos algumas iniciativas direcionadas no sentido da parametrização de unidades maiores ou da formalização de especificações de requisitos de projeto. Os estudos empreendidos por Kiviniemi (2005) e Araújo (2014), exemplificam a complexidade envolvida neste tipo de iniciativa.

Neste trabalho, descrevemos a formalização de uma estrutura de informação cujo objetivo é auxiliar em tarefas de lançamento e controle dos elementos estruturais de uma edificação. Considera-se ao edifício como um objeto paramétrico em sua totalidade. Os objetivos principais que guiam este desenvolvimento são os seguintes:

- Tornar explícita a descrição textual do edifício como apoio documental de modelagem e verificação.
- Atingir níveis crescentes de automatização nas tarefas de modelagem utilizando técnicas de parametrização.

- Reduzir a necessidade de manipular o modelo pelo toque incorporando interfaces de voz ao processo de modelagem.
- Relacionar a forma do terreno e a implantação do edifício às formas estruturais e arquitetônicas.
- Permitir que a estrutura de informação possibilite o ingresso de unidades modulares de projeto, visando à coordenação modular da edificação.
- Testar a inversão da lógica tradicional de modelagem e verificação (modelar-verificar) pela lógica de verificar necessidades e modelar corretamente, visando diminuir as inconsistências do modelo e a quantidade de interferências desde as primeiras etapas.
- Distribuir os tipos de componentes construtivos desde as primeiras fases de projeto.
- Manter controle sobre a materialidade dos componentes construtivos.

A formalização estudada aborda o problema desde o aspecto morfológico, partindo do tratamento de unidades formais simples. Não se buscou, no momento inicial, explorar complexidades geométricas nem aderir aos conceitos derivados dos novos paradigmas morfológicos da revolução da chamada "era digital" como observada e analisada por Oxman (2006). A complexidade aqui tratada está emparentada com as tarefas de articulação da quantidade de elementos e dados que devem ser gerenciados em projetos que adotam metodologia BIM. Nesse sentido, a complexidade se relaciona com a estruturação dos dados, não necessariamente aos resultados morfológicos atingidos. Entende-se que a morfologia das fachadas, por exemplo, pode herdar alguns aspectos da morfologia estrutural, que por sua vez, pode ter os seus aspectos formais derivados da forma do terreno onde está implantado ou da volumetria resultante das condicionantes de implantação. Busca-se identificar e organizar vínculos específicos e gerais que estejam presentes na estrutura formal da edificação, associando-os com algoritmos que automatizem as tarefas de modelagem e controle posterior do modelo. Em paralelo ao desenvolvimento da formalização dos dados, tem sido implementado um conjunto de componentes programados em plataforma .NET e

em linguagem C#. Os componentes cumprem a função de automatizar os processos a partir dos dados definidos. Além de perseguir a automatização integrada do processo de modelagem, a pesquisa também busca consolidar a definição de unidades semânticas que permitam integrar interfaces por voz (VUI) ao conjunto de comandos e solicitações necessárias para ativar ditos processos. Busca-se complementar os modos tradicionais de operação, hoje em dia representados pelas interfaces gráficas (GUI), às possibilidades abertas por este tipo particular de interface. O objetivo é reduzir paulatinamente a necessidade de manipulação ou toque dos elementos do modelo por parte dos usuários, face à complexidade crescente que um modelo tridimensional completo apresenta. As categorias e vínculos definidos, junto aos seus valores associados, foram estruturados, codificados e armazenados em arquivos de texto externos ao modelo. Se pretende com isto, auxiliar a todos os agentes envolvidos, desde o projetista até os gerentes e coordenadores, tornando explícito o conjunto de características que deram forma ao projeto. A técnica proposta visa a ser um auxílio no entendimento do projeto, organizando seus dados morfológicos, tanto nas etapas de concepção como nos momentos de operação. Nesse sentido, os dados referentes à posição relativa dos elementos construtivos dentro do projeto adquirem significado fundamental, como assim também os dados referentes à estruturação morfológica da edificação, como os padrões rítmicos da estrutura e as suas dimensões modulares. A decisão de manter a estrutura de dados externa ao modelo relaciona-se com as tarefas de colaboração. Para levar a cabo as diversas operações de controle, durante ou após finalizado o projeto, é fundamental que toda a lógica formal que estruturou o edifício não se perca, volatilize ou fique apenas preservada de modo implícito dentro do modelo. Nesse sentido, a função dos arquivos de texto é preservar, explicitamente, as informações que deram origem à forma do prédio. Portanto, a estrutura de informação perseguida pretende ser um resumo textual da forma atualizada do projeto.

O objeto

Na primeira etapa da pesquisa foi limitada a complexidade do problema ao estudo e formalização dos dados morfológicos dos elementos estruturais. O objeto de estudo trata de um prédio funcionalmente genérico, de planta baixa regular, compacto e com estrutura convencional de concreto. Determinou-se que a forma procurada apresentasse uma subdivisão vertical clássica, composta por fundações, subsolos, embasamento e torre. Também foi previsto a incorporação de pavimentos técnicos; a diferenciação de alturas de determinados pavimentos chaves como o térreo, cobertura e subsolos; a organização de três tipos de posicionamento para fundações, colunas, vigas e lajes canto, perimetral e interno; e, o tratamento diferenciado de tipos de lajes de acordo ao andar. Este estudo visa expandir e generalizar a estrutura para casos morfológicos e funcionais de maior diversidade e complexidade, assim como formalizar os aspectos que se interliguem a outros tipos de requisitos.

Organização dos dados

Os dados referentes ao projeto são registrados em arquivos de texto externos cujo conteúdo responde a domínios específicos. Na primeira etapa de implementação, esses domínios foram distribuídos da seguinte maneira:

- Dados_Edificio_Localizacao.txt
- Dados_Edificio_Morfologia.txt
- Dados_Edificio_Compartimentos.txt
- Dados_Edificio_Materiais.txt
- Dados_Edificio_Pisos.txt
- Dados_Edificio_Dados.txt

Os campos contidos em cada arquivo estão estruturados de modo tradicional por uma corrente de caracteres aberta na qual cada uma das partes aponta para uma categoria específica.

CHAVE_DE_ACESSO = VALOR

Em relação à localização do projeto, por exemplo, foram definidas três categorias de dados, subdivididas internamente.

PROJETO_XXXX
LOCAL_XXXX
LOTE_XXXX

----- Dados do projeto e implantação do edifício -----

PROJETO_CLIENTE = Nome do cliente
PROJETO_NOME = Nome do projeto
PROJETO_CODIGO = A001
PROJETO_TIPO = Escola
PROJETO_CLASE = Reforma
PROJETO_ANO = 2016
PROJETO_MES = 1
PROJETO_DIA = 1
PROJETO_UNIDADES = METROS

LOCAL_CIDADE = Rio de Janeiro
LOCAL_BAIRRO = Cidade Universitária
LOCAL_RUA_NOME = Av. Athos da Silveira Ramos
LOCAL_RUA_NUMERO = 149
LOCAL_RUA_COMPLEMENTO = Bloco D

LOCAL_AREA_DE_PLANEJAMENTO = AP2
LOCAL_REGIAO_ADMINISTRATIVA = RA XX
LOCAL_ZONA_ADMINISTRATIVA = ZR3

LOTE_TAXA_DE_OCUPACAO = 60
LOTE_INDICE_APROVEITAMENTO = 4
LOTE_DIVISA_FRENTE_DIM = 80.0:0
LOTE_DIVISA_FUNDO_DIM = 80.0:0
LOTE_DIVISA_DIREITA_DIM_ANG = 100.0:85
LOTE_DIVISA_ESQUERDA_DIM_ANG = 100.0:95
LOTE_GABARITO_PERMITIDO = 20

LOTE_AFASTAMENTO_DIVISA_FRONTAL = 10.0

LOTE_AFASTAMENTO_DIVISA_FUNDOS = 5.0
LOTE_AFASTAMENTO_DIVISA_LATERAL = 5.0

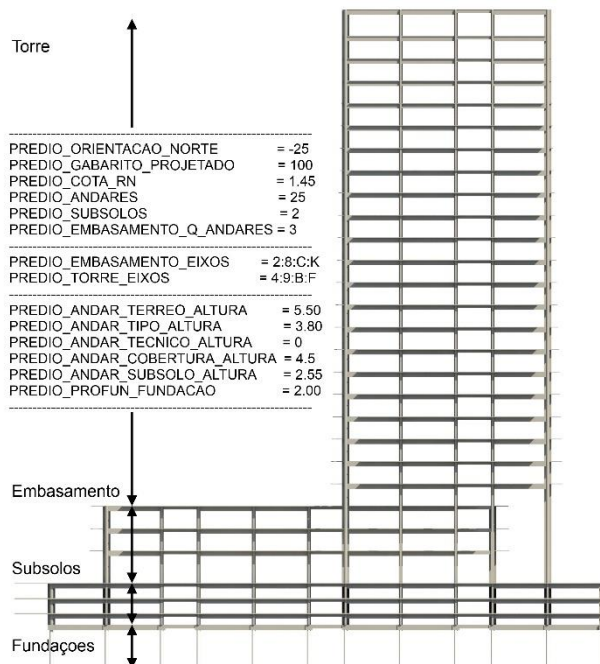


Figura 1: Dados associados às zonas verticais.

O conjunto dos arquivos de dados é armazenado no ambiente de desenvolvimento do projeto, permanecendo externo ao modelo, disponível e compartilhado pela equipe de projetistas. Os dados armazenados são utilizados para alimentar os aplicativos programados que automatizam os processos de modelagem e distribuem a informação pelos objetos. Embora nesta primeira etapa da pesquisa tenha se trabalhado apenas com a estrutura, o objetivo mais amplo é aumentar os níveis de automatização da modelagem e, principalmente, aumentar o nível de controle durante os procedimentos necessários para incorporar dados ao modelo. Em relação à incorporação de informação, são previstos modos diretos de inserção de dados, assim como modos de inserção por inferência e cruzamento dos campos. Os códigos Omniclass, Uniformat, MasterFormat são exemplos típicos. No algoritmo de modelagem são relacionados os códigos da tabela Omniclass nº12, que classifica a construção pela sua forma. Os campos que devem ser relacionados para o preenchimento da informação na estrutura proposta são:

PREDIO_ANDARES = 6
PREDIO_FORMA = ISOLADA:PONTUAL

Eles permitem que o algoritmo discrimine o preenchimento do campo correspondente ao Omniclass do modelo BIM referente à forma do projeto. Neste caso o valor assignado seria 12-11-14-14.

Distribuição no plano horizontal

A distribuição no plano horizontal do projeto foi definida considerando três aspectos:

1. A forma do terreno.
2. A distribuição rítmica dos eixos estruturais.
3. A posição relativa de cada elemento dentro do conjunto.

Algumas propriedades recebem um dado único, mas em alguns casos o valor associado à propriedade pode requerer mais de um dado. Nesses casos, os valores são englobados, concatenando-os numa correte de informação com separadores definidos por dois pontos (:) na seguinte maneira *dado1:dado2:dado3:dado4....* Visando a simplificação dos campos necessários para descrever a forma de lotes quadriláteros optou-se por concatenar os valores dimensionais e angulares das divisas, armazenando-os na forma de um par *dimensão:ângulo*.

LOTE_DIVISA_DIREITA_DIM_ANG = 100.0:85

O algoritmo que faz a leitura dos pares de valores associados a essa definição é a seguinte função:

```
public string Pega_dado(string dado, int i)
{
    return dado.Split(':')[i];
}
```

Para a distribuição rítmica dos eixos estruturais foram definidas três possibilidades, simbolicamente representadas como AAA, ABA, AAB. Esses ritmos dimensionais são aplicados de maneira independente para os eixos verticais e horizontais.

MODULOS_V_RITMO = AAB
MODULOS_H_RITMO = ABA

Os ritmos são associados a dimensões modulares específicas as quais estão, por sua vez, associadas a um módulo construtivo dimensional básico definido no projeto. Assim, a dimensão entre eixos é obtida multiplicando-se a quantidade de módulos pela dimensão do módulo dimensional básico.

MODULOS_V_QUANTIDADE_A = 12
MODULOS_V_QUANTIDADE_B = 10
MODULO_BASICO_CONSTRUCTIVO = 0.625
MODULO_BASICO_FUNCIONAL = 6Mc

Neste exemplo, o módulo funcional é representado como 6 unidades de módulo construtivo ($6 \times 0.625 = 3.75$). A definição do módulo funcional responde à necessidade de estabelecer as dimensões mínimas desejadas para os compartimentos básicos do projeto (salas, laboratórios, etc.).

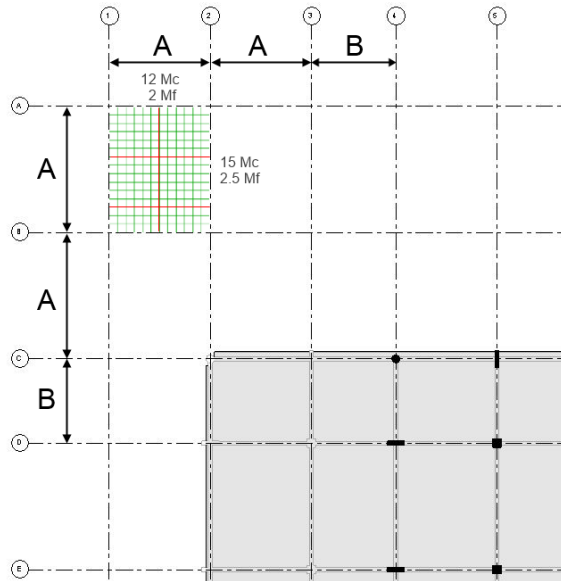


Figura 2: Divisão rítmica e modular da estrutura.

Para distribuir os elementos estruturais em altura foi estabelecida uma divisão clássica de partes específicas da edificação (Fundação, Subsolos, Térreo, Embasamento, Torre e Cobertura). Para determinar a distribuição dos elementos sobre o plano horizontal levou-se em consideração a sua posição relativa dentro da planta baixa. Foram estabelecidas três posições específicas para as fundações, colunas, vigas e lajes:

- Elemento estrutural de canto
- Elemento estrutural de perimetral
- Elemento estrutural interno

Um elemento estrutural posicionado sobre um eixo estrutural extremo é definido como perimetral. Caso esteja posicionado sobre dois eixos extremos, é definido como um elemento de canto. O algoritmo em C# utiliza uma definição “struct” para organizar e processar os dados durante a etapa de modelagem.

```
public struct Elemento_Estrutural
{
    public string Setor;
    public string Tipo;
    public string Prefixo;
    public string Forma;
    public int Andares;
    public int VariarSecao;
    public XYZ Ponto1;
    public XYZ Ponto2;
    public double AreaInfluencia;
    public double AnguloRotacao;
    public double Altura;
    public double Largura;
    public string Posicao;
    public double Parapeito;
    public double Fator_de_offset_Y;
    public ElementId Material;
    public FamilySymbol Familia;
}
```

As famílias utilizadas em cada posição são especificadas nos arquivos de dados externos, permitindo-se o uso de famílias diferenciadas. Os membros de cada tipo são criados e classificados automaticamente durante o processo algorítmico de lançamento da estrutura. Durante o processo de inserção automática, o algoritmo discrimina espacialmente qual família deve ser inserida e qual rotação deve ser aplicada de acordo à posição relativa estipulada.

COLUMNAS_FAMILIA_C = Coluna_Circular.rfa
COLUMNAS_FAMILIA_P = Coluna_Retangular.rfa
COLUMNAS_FAMILIA_I = Coluna_Retangular.rfa

Na versão atual, o algoritmo realiza o pré-dimensionamento de colunas e vigas, utilizando fatores específicos e dimensões mínimas previstos também nos arquivos de texto.

VIGAS_FATOR_PREDIMENSIONAMENTO = 12
VIGAS_FATOR_ALTURA_LARGURA = 2.5

Na definição do *struct* também foram definidos dados específicos para assignar o material ao objeto modelado, definir a nomenclatura de tipo dentro da família; determinar a sua forma; e, finalmente, o ponto coordenado de localização do elemento. Alguns elementos, como no caso das vigas, podem ter as suas dimensões associadas às dimensões dos elementos aos quais estão relacionados. Nesses casos, o valor definido aponta para o nome do campo específico.

VIGAS_DIM_LARGURA_MAXIMA = COLUMNAS_DIM_MINIMA

O lançamento de vigas foi programado para colocá-las em três posições específicas, com o valor associado de elevação quando necessário. No exemplo da figura 3, as vigas internas foram colocadas no modo normal, enquanto as vigas de canto e periféricas foram lançadas em posição semi-invertida com a definição de altura do parapeito.

VIGAS_CANTOS_POSICAO = semi-invertida:0.5
VIGAS_PERIFE_POSICAO = semi-invertida:0.5
VIGAS_INTERN_POSICAO = normal:0

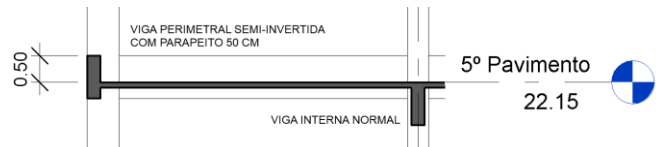


Figura 3: Posição das vigas.

Distribuição nos planos verticais

Foi estudada uma solução algorítmica para introduzir variação formal na torre e no embasamento de modo independente. Para introduzir variação formal, se utilizaram os cantos do contorno da edificação como elementos chave. Foi definido um algoritmo que regula a definição de cada canto através de nove posições numeradas e específicas, cujo ponto de ancoragem (0) é a interseção calculada dos dois eixos estruturais correspondentes ao canto estudado. Os nove pontos

mapeados permitem que cada quina do edifício seja automaticamente definida de acordo a valores formais numéricos predefinidos pela equipe de projeto. A figura 4 destaca alguns gabaritos formais possíveis.

TORRE_CANTOS_FORMA = 45078
EMBASAMENTO_CANTOS_FORMA = 31

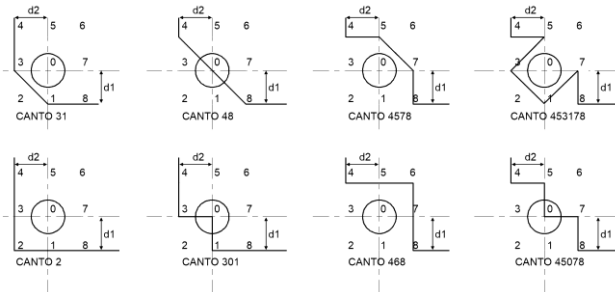


Figura 4: Soluções de contorno para os cantos do edifício.

As distâncias entre o eixo estrutural e a borda da laje (d1 e d2) permitem introduzir um movimento de inclinação vertical para cada uma das quatro fachadas de maneira independente. Os bordos podem também ser curvos ou retos.



Figura 5: Configuração da estrutura com fachadas retas.

A variabilidade formal consegue-se pelo uso diferenciado desses parâmetros que são concatenados nas diversas transformações. No momento, foram implementadas as seguintes transformações:

- Parâmetros de inclinação vertical das fachadas, tanto em sentido positivo quanto negativo pelo incremento ou decremento das distâncias d1 e d2.
- Definição de alternância vertical das transformações.

- Possibilidade de inclinar horizontalmente as quatro fachadas.
- Alteração da curvatura da fachada que pode ser côncava, convexa ou reta.

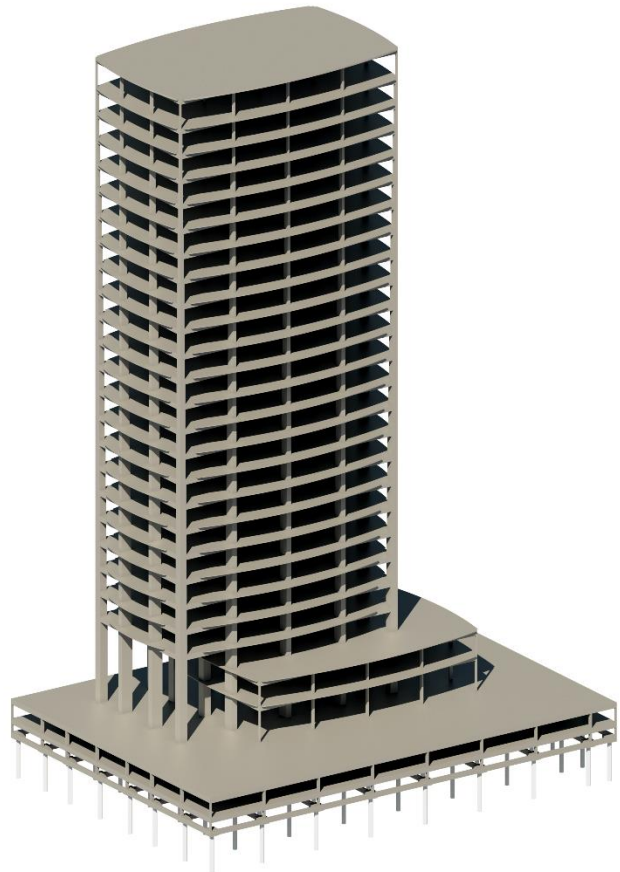


Figura 6: Configuração da estrutura com fachadas de curva convexa.

Gerenciamento de materiais

Outro aspecto estudado foi a definição da materialidade do edifício. Os dados referentes aos materiais utilizados são registrados no arquivo Dados_Edificio_Materiais.txt. Nele são distribuídos os materiais assignados para cada elemento do projeto.

MATERIAL_LAJES = M1
MATERIAL_VIGAS = M1
MATERIAL_COLUNAS = M1
MATERIAL_FUNDACOES = M1
MATERIAL_ESTACAS = M1

A lista acima é complementada relacionando outros elementos como tipos de paredes, contra pisos e forros a seus respectivos materiais. O código do material (M1 neste exemplo) é utilizado como chave do nome para acessar todas as definições especificadas. Como previsto para os outros elementos estruturais, os materiais também foram preparados com campos específicos que permitem a incorporação dos

diversos padrões de codificação que existem (Omniclass, ABNT, etc.). Embora todos possam ser preenchidos, cabe à equipe decidir quais campos serão utilizados.

```
MAT_M1_NOME = Concreto estrutural
MAT_M1_DESCRICAO = Descrição do material
MAT_M1_CLASSE = Concreto
MAT_M1_CUSTO = 1100:m3
MAT_M1_FABRICANTE = Nome do fabricante
MAT_M1_DADO_MASTERFORMAT = cod_Masterformat
MAT_M1_DADO_OMNICLAS = cod_Omniclass
MAT_M1_DADO_UNIFORMAT = codigo_Uniformat
MAT_M1_DADO_SINAPI = codigo_Sinapi
MAT_M1_DADO_ABNT = código_ABNT
```

As definições de renderização, sombreado, propriedades mecânicas e térmicas são acrescentadas nesta sequência de dados. No caso das lajes, quando compostas por diversas camadas, o algoritmo de criação encarrega-se de montá-las definindo as suas espessuras e materiais.

```
Laje_Tipo_1_A1 = M1:0
Laje_Tipo_1_S1 = M1:0
Laje_Tipo_1_ES = M1:0.10:Convencional
Laje_Tipo_1_S2 = M1:0
Laje_Tipo_1_A2 = M1:0
```

No exemplo acima, a Laje Tipo 1 possui apenas uma camada estrutural composta pelo material M1, com espessura de 10 cm e classificada como tipo convencional.

A interface VUI

Kiviniemi observa que a complexidade de um modelo de formalização de especificação de requisitos de um edifício pode ser gerenciada por uma interface de usuário bem projetada (Kiviniemi, 2005:109). É, nesse sentido, que nesta experiência destaca-se a importância de relacionar a proposição formal da estrutura de dados à formalização das interfaces de usuário destinadas a gerenciar tais dados. Tenta-se manter ambos os aspectos coordenados. Um dos objetivos do trabalho é produzir observações que conduzam a interfaces que permitam minimizar a necessidade de manuseio do modelo. As interfaces direcionadas por voz podem contribuir a cumprir com esse objetivo. Embora seja uma tecnologia que teve seus inícios na década de 1950, somente em época recente o Reconhecimento de Voz (*Speech Recognition* SR) vem se tornando cada vez mais presente nos dispositivos de comunicação utilizados. A tecnologia admite dois modos de uso: por ditado, o que exige a criação de um banco de dados no qual ficam registrados os perfis sonoros do timbre de voz dos usuários; ou, o reconhecimento dirigido por gramáticas previamente definidas, o que exige criar e registrar em arquivos específicos um léxico bem estruturado. A presente proposta concentra-se no reconhecimento dirigido por gramáticas. O projeto da estrutura gramatical deve conter três elementos básicos:

1. Léxico de entrada reconhecido pelo mecanismo SR;
2. As regras das combinações lexicais;

3. As saídas que serão repassadas para as aplicações.

As interfaces de voz permitem programar definições textuais que serão interpretadas pelo mecanismo de reconhecimento. Cada regra programada deve ter assignado um nome e deve conter registradas todas as locuções que ao serem vocalizadas serão reconhecidas.

```
<rule id="Nome da regra">
  <item>
    <one-of>
      <item>Locução 1</item>
      <item>Locução 2</item>
    </one-of>
  </item>
</rule>
```

Como diversas locuções podem ser adicionadas a uma regra, é possível executar um comando realizando a solicitação falada de várias maneiras diferentes, extraindo uma ordem através de uma saída definida com <tag>out = "ordem"</tag>. As regras e as saídas podem ser re combinadas por concatenação para formar um domínio lexical específico de modelagem e gerenciamento do projeto. Portanto, as definições semânticas estão intimamente relacionadas tanto à estrutura de dados definida e utilizada pelo algoritmo executor, como às maneiras em que os usuários se referirão conceitualmente aos elementos construtivos ou às diversas situações e componentes da edificação. A definição de um canto chanfrado, segundo a estrutura de dados estudada, pode ser estabelecida pela seguinte regra.

```
<rule id=" forma do canto ">
  <item>
    <one-of>
      <item> Chanfro externo</item>
      <item> Chanfrado exterior</item>
      <item> Chanfre por fora</item>
      <item> Chanfrar por fora</item>
      ....
    </one-of>
  </item> <tag> out = "31" </tag>
</item>
</rule>
```

O valor de saída desta requisição (31) é referente a um dos tipos de cantos estudados.

TORRE_CANTOS_FORMA = 31

A técnica de aplicar comandos falados foi estudada numa pesquisa anterior. Nessa oportunidade, foi programado um aplicativo de modelagem em ambiente Revit. O aplicativo contém um léxico de comandos que permite executar as operações de modelagem acionadas pela requisição vocal do usuário. Experimentando com o sistema chegou-se à conclusão de que esse tipo de interface pode ser eficiente quando utilizada para ativar rapidamente operações de configuração geral de ambiente de trabalho (seleção de unidades, controle de zoom, seleção de tipos de sombreado, definição horária para estudos solares, etc.).

Uma vantagem que a interface por voz proporciona é a naturalidade das solicitações, permitindo que o usuário tenha cada vez menos necessidade de memorizar a localização dos elementos de acionamento da interface, como botões, menus ou quadros, presentes nas interfaces gráficas (GUI). No entanto, verificou-se que ao ser utilizada para executar operações específicas de traçado, na interface por voz existe um problema a ser superado: a interferência do impulso gestual presente nessas operações, uma vez que a ação de desenhar ou modelar é uma questão de interação de ordem gestual. Daí a necessidade de se definir um domínio semântico mais específico e consistente, que se caracterize por ser sistemático, mas que ao mesmo tempo não se afaste da linguagem natural, visando a integração das definições construtivas e formais numa semântica de dados rica e facilmente vocalizável. Busca-se ampliar o uso deste tipo de interface para além das operações de configuração geral, facilitando também as operações de criação e edição do modelo. A hipótese deste trabalho assume que utilizando um modelo organizado dentro de um domínio estruturado de informações formais, bem delimitado, seja possível tornar mais eficiente a solicitação falada de elementos construtivos em modelos complexos, evitando-se a necessidade de manipular os objetos através do toque. Em outras palavras, aumentar o aspecto puramente intelectual da tarefa de projetar, diminuindo os aspectos sensitivos. Visa-se, como objetivo secundário, reduzir a quantidade de elementos das interfaces gráficas, com a intenção de minimizar esforços cognitivos desnecessários associados à memorização da posição desses elementos de interface, dando mais tempo para a reflexão sobre os aspectos do projeto.

Selecionar pilares da fachada Norte.

Apagar colunas internas do primeiro pavimento.

Informar volume de concreto do projeto.

Apagar as colunas internas do primeiro pavimento.

Cambiar coluna B5 por coluna tipo X.

As declarações acima são passíveis de serem definidas na interface de comandos de voz. Elas ilustram o tipo de requisições e interações faladas que podem ser disponibilizadas para os projetistas.

Considerações finais

A estrutura de dados testada na experiência não pretende ser um substitutivo dos tradicionais modelos de dados BIM, sejam estes proprietários ou neutros como o IFC. Ela está sendo concebida para funcionar como uma camada adicional de dados, paralela e complementar. A formalização proposta para o lançamento automático da estrutura, permitiu obter um gerenciamento ordenado dos elementos incorporados ao projeto. Embora o método apresentado possa ser utilizado para testar diversas hipóteses formais de um projeto, o seu domínio de busca combinatória ainda é restrito a prédios de planta compacta. O leque de possibilidades formais alcançado limita-se às combinações permitidas pelo algoritmo, que pode ser entendido como um algoritmo determinista. O método proposto pode ser integrado no gerenciamento do projeto em etapas avançadas, quando as mudanças de volumetria ou de

distribuição de layout passam a ser menos radicais, tendo o projeto estabilizado numa forma mais ou menos definitiva. Os arquivos externos contendo as definições formais adotadas servem de memória documental de consulta do projeto e verificação do modelo. O conjunto formado pelo par “dado – algoritmo” torna-se o sustento que permite criar e reformular automaticamente o modelo virtual de maneira precisa e consistente. Resta ainda um longo caminho a percorrer para uma formalização completa. Os próximos passos incluem consolidar a formalização geral da estrutura agregando-lhe novas definições, como por exemplo a incorporação dos dados de carga da estrutura, visando a sua exportação para um sistema de análise estrutural. A pesquisa pretende avançar em outras frentes, que vão desde a incorporação de automatizações adicionais que ampliem as possibilidades combinatórias de organização dos elementos estruturais, assim como a incorporação de novos elementos arquitetônicos. Paralelo ao aprimoramento da versão atual do aplicativo, será iniciada uma próxima etapa na qual se estudará a incorporação dos elementos de fechamento das fachadas articulando-os com a estrutura. Uma consideração importante é que a automatização apresentada neste trabalho não é fechada em si mesma, pois permite que os usuários possam realizar alterações da estrutura. Para garantir a consistência dos dados, estão sendo implementadas funções para manter sincronizadas as alterações do modelo com os arquivos de dados externos.

Referências

- Araújo P. S. M. S. (2014). Contribuição do BIM no processo de projeto de arquitetura. Uma aplicação ao projeto de biotérios. 2014. Tese de doutorado. Universidade Federal Fluminense. Niterói.
- Ilhan B., Yaman H. (2015). IFC-Based Sustainable Construction: BIM and Green Building Integration Proc. 32nd CIB W78 Conference 2015. Eindhoven, Netherlands.
- Kiviniemi, A. (2005). Requirements management interface to building product models. CIFE Technical Report #161. Stanford University.
- Krieger J. (2013, Apr). My BIM journey. 6 lessons from a BIM/VDC expert. Abril, 2013. Building Design+Construction. Illinois, USA.
- Menegotto, J.L. (2015). A Framework for Speech-Oriented CAD and BIM Systems. In G. Celani, D. M. Sperling, J. M. S. Franco (Eds.), Computer-Aided Architectural Design Futures. The Next City - New Technologies and the Future of the Built Environment (pp. 329-347). Springer Berlin Heidelberg.
- Oxman, R. (2006). Theory and design in the first digital age. Design Studies, Vol 27, N° 3.
- Oxman, R., Oxman, R. (2010, Jul.). The new structuralism: design, engineering and architectural technologies. AD Vol. 80, I. 4. (pp. 15-23).
- Venugopal M., Eastman C.M., Sacks R., Teizer J. (2012). Semantics of model views for information exchanges using the industry foundation class schema. In Advanced Engineering Informatics Vol. 26 I. 2, (pp. 411- 428).
- Voice Browser Working Group: Speech Recognition Grammar Specification Version 1.0 W3C. Retrieved from: <http://www.w3.org/TR/speech-grammar>