# Research on CAD-Based BIM Model Axis Grid Generation Method Postprint

**Authors:** Ma Cheng (1); Lu Xiaolong (2); Wang Dejiang (1)

**Date:** 2017-11-06T00:00:00+00:00

## Abstract

This paper analyzes DXF files generated from CAD drawings to identify grid lines and annotations. Simultaneously, employing Revit API development technology and leveraging the extracted CAD information, it automatically generates corresponding grid lines and annotations in the Revit model, thereby reducing the manual effort currently required for constructing BIM models from CAD drawings.

## Full Text

## Preamble

**Vol. 8 No. 1, February 2016**

**Research on Axis Generation Method for BIM Models Based on CAD Drawings**

Ma Cheng[1], Lu Xiaolong[2], Wang Dejiang[1]
(1. Department of Civil Engineering, Shanghai University, Shanghai 200072, China;
2. State Grid Shanghai Municipal Electric Power Company Economic and Technology Research Institute, Shanghai 20000, China)

**Abstract:** This paper analyzes DXF files generated from CAD drawings to identify grid lines and their annotations in the drawings. Simultaneously, using Revit secondary development technology, the program automatically generates corresponding grids and annotations in the Revit model based on the extracted CAD information, thereby reducing the current manual workload of building BIM models from CAD drawings.

**Keywords:** BIM; CAD; Secondary Development
**CLC Number:** TU17
**Document Code:** A

**Article ID:** 1674-7461(2016)01-0043-05
**DOI:** 10.16670/j.cnki.cn11-5823/tu.2016.01.07

## Introduction

BIM (Building Information Modeling) technology, which integrates various types of building information, is poised for widespread application in civil engineering design, construction, operation, and maintenance. Consequently, it has received substantial promotion and adoption both in China and worldwide. The United States has continuously updated its national BIM standards, currently releasing its third edition, while the UK government mandates BIM technology for all government procurement projects. In China, the Ministry of Housing and Urban-Rural Development and various provinces have successively issued development guidelines for construction informatization.

Currently, due to limitations in civil industry specialization and the fact that most designers are accustomed to CAD-based design, domestic BIM model construction typically occurs after design institutes produce drawings. BIM professionals then manually build BIM models based on CAD drawings, a process that involves substantial workload. How to rapidly generate BIM models from CAD drawings is an urgent problem requiring solution.

In BIM design software, Autodesk' s Revit series products dominate the civil building sector due to the widespread use of the company' s CAD software. Revit provides powerful secondary development interfaces, namely the API (Application Programming Interface), which makes it possible to enhance Revit functionality through programming.

This paper proposes a method that reads DXF files exported from CAD to identify grid lines and annotations in drawings. Using Revit secondary development technology, the method automatically generates BIM model grids and annotations based on CAD drawing information, providing a foundation for rapid modeling and reducing manual workload.

## 1. Development Tools and Drawing Standards

### 1.1 Development Tools

This research adopts Visual Studio 2010 as the programming tool, with C# as the programming language and Revit 2014 as the software platform. Revit' s secondary development utilizes the Revit SDK, which provides extensive Revit interfaces.

During development, since the information read through secondary development is Revit' s internal data rather than interface information, it is necessary to use Revit's Revit Lookup plugin to examine all properties of various family instances, as shown in Figure 1.

### 1.2 CAD Drawing Standards

To facilitate the identification of axis and grid annotation information in CAD drawings and ensure that the generated axes in Revit maintain consistency with those in CAD in terms of length, spacing, and coordinates, CAD drawings must be pre-modified according to established standards, with all axes placed on a single layer.

## 2. DXF File

### 2.1 DXF File Structure

This paper adopts the method of reading DXF files exported from CAD to access CAD drawing content. A DXF file consists of paired integer codes and associated values, which CAD refers to as group codes and group values respectively, each occupying one line. A complete DXF file comprises six sections: HEADER, CLASSES, TABLES, BLOCKS, ENTITIES, and OBJECTS. Each section begins with SECTION and ends with ENDSEC. The composition and meanings of some group codes are shown in the table below. This paper focuses on the ENTITIES section, which contains various entities and any block references.

### 2.2 DXF File Reading

In CAD drawings, grids are composed of lines, circles, and text. As previously described, DXF files consist of group codes and group values. When CAD graphics are saved as DXF files, these graphics are represented by coordinates, radii, and text heights. For different entities such as circles and lines, different reading methods must be programmed to store their respective parameters. The program can loop through and determine data types based on entity names and layer names. The flowchart for reading DXF files is shown in Figure 3, and the complete process is illustrated in Figure 4.

## 3. BIM Model Grid Generation

### 3.1 Grid Generation

The class corresponding to grids in Revit is Grid, which inherits from Element. The grid creation function in Revit is NewGrid, which has two overloads for lines and curves respectively:

```
Document.Create.NewGrid(Arc)
Document.Create.NewGrid(Line)
```

Grid establishment can be achieved by creating curves and lines separately and then substituting them as parameters into the NewGrid method. Using this approach, grids identical to those in CAD can be generated in Revit based on data read from DXF files. The methods for creating lines and curves are as follows:

```
Line geomLine = Line.CreateBound(start, end)
Arc geomArc = Arc.Create(START, END, ZHONG)
```

### 3.2 Grid Annotation Modification

When generating grids in Revit, the annotation issue must be addressed. Revit automatically names grids during creation—for example, the first grid is named "1" or "A" , with subsequent grids incrementing sequentially. This automatic naming cannot meet the requirements for correspondence with CAD drawing annotations. The implementation method in this paper involves finding circles that intersect with grid extension lines or tangents, reading the text within the circles, and assigning it to the grids. In Revit, grid names can be modified through the Grid.Name property. The grid reading process is shown in Figure 5, and the complete grid creation process is illustrated in Figure 6.

## 4. Research Example

To validate the proposed method, a building grid example was imported into Revit. The CAD grid drawing is shown in Figure 7, containing 6 vertical grids, 4 horizontal grids, and 1 curved grid. All grids are black with default line types.

The grids generated in Revit are shown in Figure 7. The generation method uses external commands loaded into Revit through AddinManager, an official Autodesk plugin for loading Revit plugins. AddinManager allows plugin code modification and reloading without restarting Revit. Since grids are generated directly from endpoint coordinates, the insertion positions are completely guaranteed to be accurate.

The generated grids maintain complete consistency with those in CAD in terms of insertion positions and annotations. However, line types and colors have not been considered and will be improved in subsequent research. Grids are only the foundation; future research will incorporate columns and beams to complete the import of entire frame structures.

## Conclusion

This paper achieves identification of grids and annotations from CAD drawings through DXF file reading and Revit secondary development technology, enabling automatic generation of axes and annotations in Revit based on recognized CAD information. The paper briefly describes DXF data format and key technologies for identifying axes from DXF files, and details the implementation of using Revit secondary development technology for grid generation and annotation in Revit, solving the correspondence problem between BIM model grids and CAD drawings. Conclusions are as follows:

1. Using Revit secondary development technology to automatically generate BIM model grids is feasible.

2. The integration between CAD drawings and BIM models can be achieved through DXF as an intermediate file.

The generated grids maintain positional and annotation consistency with CAD drawings, though line type and color issues remain to be addressed in future research.

## References

[1] He Qinghua, Qian Lili, Duan Yunfeng, Li Yongkui. Research on Current Status and Obstacles of BIM Application at Home and Abroad[J]. Journal of Engineering Management, 2012(1): 12-16.

[2] Zhang Jianping, Li Ding, Lin Jiarui, Yan Gangwen. Application of BIM in Engineering Construction[J]. Construction Technology, 2012, 16: 10-17.

[3] He Guanpei. BIM and BIM-related Software[J]. Journal of Information Technology in Civil Engineering and Architecture, 2010, 2(4): 110-117.

[4] He Lingtong. Current Status of BIM Application Worldwide[J]. Engineering Quality, 2013, 03: 12-19.

[5] Chris Moore, Jeffrey W. Oullet, Wang Na. U.S. National BIM Standard Version 3 Promotes Progress in A/E/C/O Fields[J]. Journal of Information Technology in Civil Engineering and Architecture, 2014, 6(2): 20-23.

[6] Xu Jian. Research on Secondary Development of Revit System Software[J]. Railway Technical Innovation, 2014, 05: 39-41.

[7] Wang Jianyu, Wang Xinyan. Research on Rapid Modeling Technology from AUTOCAD to REVIT through Secondary Development[J]. Journal of Information Technology in Civil Engineering and Architecture, 2015, 7(3): 111-115.

[8] Liu Chuanliang, Lu Jiande. AutoCAD DXF File Format and Secondary Development Graphics Software Programming[J]. Microcomputer Development, 2004, 09: 101-104.

[9] Yao Yibin, Kong Jian. Research and Program Implementation of Drawing Conversion Method Based on DXF File[J]. Journal of Geodesy and Geodynamics, 2011, 01: 117-122.

[10] Li Fangzhen, Xu Lunhui. Research on DXF File Format and Its External Interface[J]. Ordnance Industry Automation, 2008, 07: 83-85.

[11] Zhang Chengcai, Sun Ximei, Zhu Taoye. Research on DXF File Format of AutoCAD and Its Conversion Interface[J]. Microcomputer Applications, 2001, 08.

*Note: Figure translations are in progress. See original paper for figures.*