GENERATING FIRE DYNAMICS SIMULATOR GEOMETRICAL INPUT USING AN IFC-BASED BUILDING INFORMATION MODEL

SUBMITTED: March 2007
REVISED: September 2007
PUBLISHED: October 2007 at http://itcon.org/2007/29/
EDITOR: G. Lee

Johannes A W Dimyadi
AstraVision-Solutions, Auckland, New Zealand
email: jdimyadi@gmail.com

Michael Spearpoint, New Zealand Fire Service Commission Senior Lecturer
Department of Civil Engineering, University of Canterbury, New Zealand
email: michael.spearpoint@canterbury.ac.nz

Robert Amor, Associate Professor
Department of Computer Science, University of Auckland, New Zealand
email: trebor@cs.auckland.ac.nz

SUMMARY: *Fire Dynamics Simulator (FDS) is an advanced simulation tool used by fire engineers. The work described in this paper enables geometrical information to be transferred to the simulation tool using the IFC building information model. A parser tool has been developed to extract fire engineering related information from the IFC model and a web-based application has been created to generate FDS input files. Several test case geometries have been used to verify the current capability of the transfer process.*

KEYWORDS: *Fire engineering, computational fluid dynamics, geometry, building information modelling, Industry Foundation Classes.*

1. INTRODUCTION

1.1 Background

The availability and accessibility of faster computer processors with larger memory capacity makes it practical for fire engineers to use sophisticated and computationally intensive numerical simulation tools to solve complex fire engineering problems. However, the current fire modelling practice often uses a paper-based approach in gathering basic building geometry information which contributes to high overhead costs in preparation for fire simulations. The conventional fire modelling process typically starts with the manual gathering of building geometry and other building information from either printed plans or electronic Computer Aided Design (CAD) drawing files. The information must then be transcribed either manually or electronically and used to construct a set of data and instructions in the format recognised by fire simulation software tools. Each software tool may have different requirements on how to represent the building geometry.

The adoption of CAD by the architecture, engineering, construction, and facilities management (AEC/FM) industry became widespread following the release of commercial CAD products in the early 1980s. CAD data generally consists of 2D and 3D entities representing points, lines, arcs, polygons, circles, text as well as regions and solid objects. Advanced CAD geometrical objects may also contain limited set of attributes and data with user-defined values. Files produced by commercial applications are generally in proprietary formats, although common formats such as DXF and IGES have been introduced to allow data exchange between different versions of the application as well as with external applications (Amor 1997). In some cases the DXF format has been used to exchange 2D building layout as image overlays onto which other data and information are superimposed. Examples of application using this DXF data exchange method in fire engineering include Simulex (Thompson and Marchant 1995) for evacuation simulations and SMARTFIRE (Frost 2001) for fire simulations. The use of DXF, however, often requires that the source data extracted from the building model contains certain compatible types of entity and are constructed in specific manners to be interpreted correctly. For example, Simulex would not interpret walls correctly if they are exchanged as polylines rather than as separate line entities. There have also been other third-party software utilities, add-ons and plug-ins available to allow CAD data to be accessed and exchanged by external applications. The data exchange process generally involves retrieving predefined attribute values of entities and exporting them to a database table or a spreadsheet or direct linking them with the external applications. For example, this method of data sharing has been used in HVAC and fire sprinklers design packages (http://www.elitesoft.com).

For the past decade, AEC/FM industry specific CAD technology has evolved into the new generation of building modelling concept often known in the industry as Building Information Modelling (BIM) (IAI 2006). BIM can be viewed as a collaborative object-oriented modelling extension to CAD. It integrates all information related to the building in one virtual information model that can be maintained throughout the whole life-cycle of the building. The concept of BIM differs from the conventional CAD in the approach to the design process. BIM puts an emphasis on creating and managing a richer set of project information in one consolidated data repository representing the virtual building model that can be shared by all the project stakeholders. The graphical representation is merely one aspect of BIM.

It would appear that BIM offers an alternative to the current paper-based or DXF transfer methods used by fire engineers wishing to carry out fire simulations. In the work presented here a specific fire engineering tool has been selected and in particular the focus is on the transfer of geometrical information. Further details about this work are presented elsewhere (Dimyadi 2007).

1.2 IFC

Industry Foundation Classes (IFC) is a standardised non-proprietary digital building information model. It has now been put forward as an international standard (Liebich 2002) and has started to be adopted and utilised by various disciplines within the AEC/FM community. Since its initial release, the IFC data model has progressively been improved to incorporate more aspects of buildings and project information. The current IFC 2x3 version of the model (IAI, 2006) contains 653 entities and over 300 supplementary data types as well as extensible property sets (Liebich 2006). The additions and revised entities in the latest version of IFC do not have any impact on the objects exchanged in this project and so IFC 2x2 has been adopted here.

IFC model data can be exchanged between IFC-compliant software applications using STEP (Standard for Exchange of Product Data) Part 21 physical file (ISO 10303-21 2002). Due to the current variations in the modelling standards and IFC export implementations adopted by different BIM authoring software packages, the same building model may be exchanged slightly differently in the STEP files. Parser software applications must be able to cope with these variations and interpret the model correctly as intended. The observations and analyses of the IFC data and mapping carried out in this project are based on the BIM and IFC implementation in the Autodesk's Revit Building 9 software package and the corresponding STEP files it generates.

1.3 Surface and solid modelling

Surface modelling is most useful in the creation of open freeform surfaces and objects with complex shapes and this technique can also be used in some CAD/BIM software applications to construct basic geometrical objects with flat or faceted surfaces. 3D geometrically closed objects are created by defining the shell or the boundary surfaces enclosing the volume of the objects. The technique focuses on the accurate geometric representation of the object rather than concerns about its physical properties. There are a number of different standards of 3D flat surface entities recognised and in use, namely POLYFACE, REGION and 3DFACE.

Solid modelling may be a more appropriate technique in representing 3D objects in certain applications particularly when the objects' physical properties are relevant. The two most common techniques used in solid modelling are Constructive Solid Geometry (CSG) and Boundary Representation (BREP). In CSG, a solid object representing a building element such as a wall is normally constructed by the extrusion of a rectangular surface object or region in the X, Y or Z axes. Editing of solids is achieved by Boolean operations such as union, subtract, and intersect, etc. with other solid primitives. BREP represents a solid by a series of connected boundary surface elements, which are defined by vertexes, edges and loops. There are several solid modelling software engines available that are commonly incorporated into commercial CAD/BIM applications. Among these engines, 3D ACIS modeller is one of the most popular engines in use.

2. FIRE DYNAMICS SIMULATOR

2.1 Computational approach

Fire Dynamics Simulator (FDS) is a computational fluid dynamics (CFD) simulation of fire-driven fluid flow. CFD typically divides the space of interest into a large number of discreet three-dimensional control volumes for which the fundamental equations governing the conservation of mass, species, temperature, velocity and density are solved (Cox and Kumar 2002). The FDS model numerically solves a form of the Navier-Stokes equations appropriate for low-speed, thermally-driven flow with an emphasis on smoke and heat transport from fires (McGrattan 2005). Although a Direct Numerical Simulation (DNS) technique can be employed under sufficiently fine grid conditions, the default computational mode uses the Smagorinsky form of Large-Eddy Simulation (LES) technique to solve large scale hydrodynamic turbulence, a condition that typically occurs in fires. Sub-models are employed to deal with other fire related phenomena such as radiative and convective heat transfer, detector activation and sprinkler sprays. The first official version of FDS was publicly released in 2000, although it existed previously in various forms since 1980 (McGrattan 2005) and the current version of FDS is Version 4.0.7, released in March 2006.

FDS is a command-line software application written in Fortran 90 without a graphical user interface and currently available in compiled versions for Windows as well as Unix/Linux operating systems. FDS requires an input file for each fire scenario in a simple text format to represent the building geometry, materials specification, computational scope, grid resolution, boundary conditions, design fire and energy source parameters, and specifications of fire safety and mechanical systems, as well as the simulation output types.

The FDS computational domain is user defined and usually represents the overall physical bounding box enclosing a building or zone of interest. The computational domain is made up of one or more rectangular meshes, each with its own three-dimensional rectilinear grid system. The resolution of the grid is also user-specified in the input file, but the grid dimensions must meet the criteria for the Poisson solver, i.e. it must be in the form 2^l 3^m 5^n, where l, m and n are integers (McGrattan and Forney 2005). All solid obstructions and voids are forced to conform to the numerical grids.

2.2 Building element representation

Building enclosure elements and solid objects are specified in the FDS input file as a series of orthogonal solid rectangular blocks or cuboids (or right parallelepipeds) representing flow obstacles, whereas doors and windows are viewed as holes allowing fluid and particles to flow through. As each solid rectangular obstruction block must be orthogonal with respect to each other, an inclined or diagonal wall or roof must be modelled in a stair-stepping manner conforming to the grid cells.

For an obstruction group of input data, the parameters required are the sextuplet of coordinates defining the lower and upper bound of the cuboids, followed by the surface type and material definition. A void in an obstruction block is specified with similar parameters to the obstruction group. Any solid grid cells within the volume specified by the void sextuplet are removed and the obstructions intersecting the volume are automatically broken up into smaller blocks (Fig. 1).

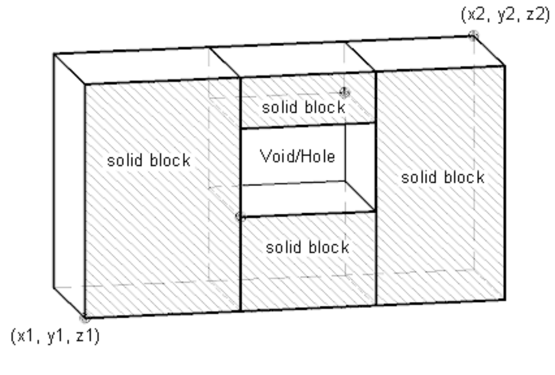


FIG. 1: A void in a solid obstruction block

Parameters can be used in conjunction with a void to specify the time at which the voids are to be created or removed, simulating the opening and closing of doors or windows. Specifying a zero time is equivalent to filling up the hole, representing a door or window that is to be kept closed for the duration of the simulation.

2.3 Creating FDS input data

The creation of the FDS input file is a part of the fire modelling process and requires varying degrees of manual input and editing particularly when considering multiple fire scenarios. Generally, the most time consuming part of the input data creation is the transfer of building geometry information from paper or CAD drawings to the format required by FDS. At the most laborious level, all the coordinates for the obstruction blocks representing the building geometry are manually determined by measurements and calculations from printed drawings. This process is particularly inefficient and error prone.

There are software tools currently available to assist with the creation of the FDS input data, particularly with respect to the transfer of the building geometry and topology information. However these tools require the reconstruction of the building in one form or another, using information derived from printed plans or CAD files. The software tools available are either only compatible with specific CAD output data formats, or they require manual data entry of the building geometry and topology information from scratch in a proprietary format.

DXF2FDS (NIST 2006) reads a DXF output created by a specific CAD application and generates the FDS equivalent obstruction blocks representing the building geometry. It can also incorporate a set of pre-defined FDS input parameters such as the computational domain and grid sizes, selected surface materials, prescribed fire source, etc. DXF2FDS has a limitation in that it only reads the 3DFACE entities and ignores all others from the DXF source file. A model constructed using other solid modelling techniques often cannot easily be converted directly into 3DFACE entities. Therefore, a tedious conversion process is often required or, alternatively, the model needs to be reconstructed using 3DFACE entities exclusively.

PyroSim (Thunderhead Engineering Consultants 2006) has been developed to construct, read and edit FDS input files. To assist with the construction of the 3D building model similar to the obstruction blocks in FDS, a 2D image of the building plan can be overlaid on the graphics editor screen to allow three-dimensional wall elements to be manually positioned by tracing over the lines. Stair-stepping is automatically applied to diagonal or curve walls to conform to the numerical grid system.

There are significant benefits in sharing a standardised digital representation of buildings in which common information can be mapped across various formats and for different input requirements of fire simulation tools. It is believed that there have not been any successful attempts to utilise the IFC data model to generate FDS input data, although NIST has indicated the intention to explore the possibility of undertaking such a project (AIA Building Connections 2006).

2.4 FDS output

The simulation outputs from FDS can be visualised graphically in an interactive 3D environment using a companion application called SmokeView (Forney and McGrattan, 2006). FDS can produce graphical output files containing the 3D model geometry, animated quantities per unit time (i.e. isosurface, slice or boundary files) as well as static pictures of the flow field in Plot3D format which can be accessed and visualised using SmokeView. The type of output quantities include space and boundary conditions such as the temperature profile, heat flux and mass flow rate, the history of heat release rates, the levels of oxygen, carbon dioxide and carbon monoxide, and other combustion products as specified. Fig. 2 shows a SmokeView snapshot of a space temperature profile (indicated by a colour spectrum) at a particular time in the FDS simulation of a test case model.

At the start of each FDS simulation, a SmokeView output file is first created containing the building geometry visualisation data. The duration of the simulation can be set to zero initially which prevents any computations from taking place except for the generation of the building geometry data. This enables the building model specified in the input file to be previewed and verified before any intensive and time-consuming computations take place.

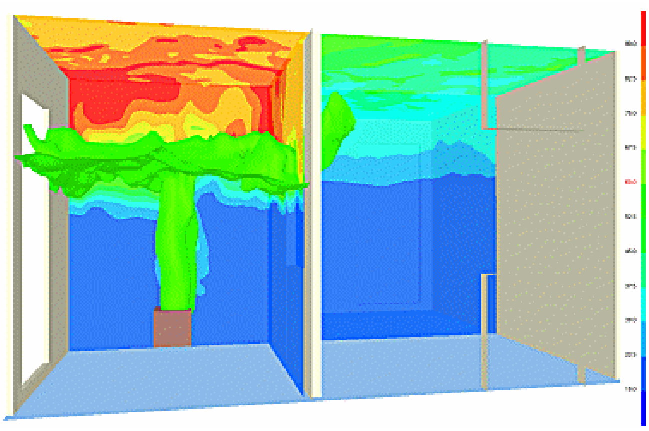


FIG. 2: Space temperature snapshot of a test case model

3. INFORMATION PROCESSING

3.1 IFC Parser for fire engineering applications

A software application is being developed to parse STEP files conforming to the IFC model and to generate a relevant subset of fire engineering entities (Spearpoint 2005). This output can then be used by software interfaces to create specific input data required by various fire simulation tools. The dedicated IFC parser for the fire engineering domain improves data access efficiency and maintains a consistent IFC version support. Software interfaces developed for various fire simulation applications only need to access the most relevant subset of IFC entities provided by the parser rather than processing the whole IFC model directly. Any revisions to the IFC model and modifications to the IFC entities only need to be implemented in the parser, which maintains a consistent common output file for use by all the software interfaces (Fig. 3). Additionally, the parser can be linked to a repository of fire engineering domain properties sets to ensure that any supported property sets incorporated into the IFC model are mapped correctly to the target fire simulation software application.

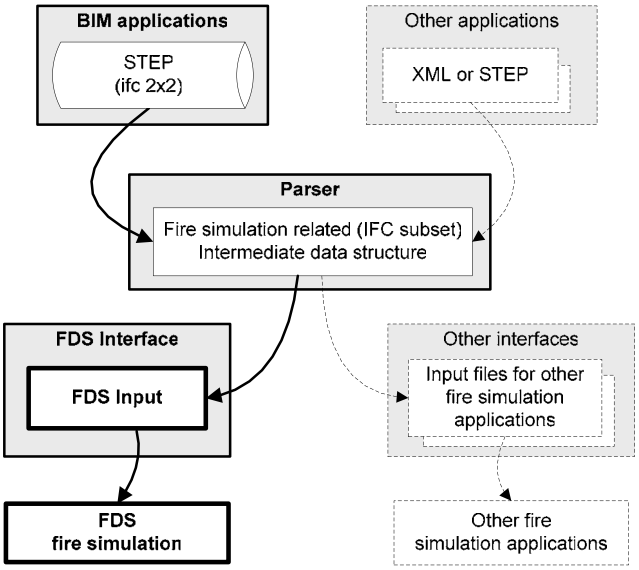


FIG. 3: Data exchange process

The current version of the parser supports IFC 2x2 and although the scope of the data mapping implementation is quite limited, it can be used to generate the input data required by BRANZFIRE zone-model fire simulation application (Wade 2003). At the outset of this work, the scope of the IFC data mapping implementation in the parser was limited to providing a set of building elements (i.e. walls, doors, windows, openings, slab) with dimensions as exchanged through STEP files exported by the ArchiCAD software package. This output was sufficient for the purposes of generating the BRANZFIRE input data as this simulation tool only requires the volume of the spaces and the area of the openings rather than the absolute positioning of the walls and openings.

Absolute positioning information of objects is required by FDS, particularly with respect to the walls and wall openings. Therefore, a wider scope of information was required for mapping the building geometry to the representation expected by FDS. The main limitations of the initial version of the parser for the purposes of generating the FDS input data were therefore that the direction of walls, door, window and wall openings were not mapped. Also the placement coordinates were mapped directly from the STEP file without modifications for aligning with a standard reference axis. A convention was necessary to accommodate for different cases due to the variants in modelling and implementation. Finally, only *IfcWallStandardCase* entities were supported and, apart from the *IfcSite* and *IfcBuildingStorey* entities, the *IfcSpace* entity was expected in the STEP file, which was required by the BRANZFIRE interface to the fire engineering data subset (Spearpoint 2005).

To extend the scope of the information exchanged, some changes and additions to the mapping implementation in the parser were required by establishing a convention for the placement of walls and wall openings by referencing the placement coordinates on the wall centreline of the axis. A convention for the placement of wall openings by realigning the opening direction with that of the parent wall and transforming the placement coordinates accordingly was also needed. Methods to determine wall heights for non-standard walls also had to be investigated.

Autodesk's Revit Building 9 software package was chosen as the BIM authoring tool to construct the test case models used in this work. It was considered worthwhile using another industry's leading BIM product as an alternative to ArchiCAD used previously (Spearpoint 2005). The IFC conforming STEP files of the models are obtained by a direct export from Revit Building. The IFC export function of the software package caters for three different IFC formats, namely IFC2x2, IFC2x3, and IFC BCA ePlan Check intended for the ePlan submission required by the Building and Construction Authority of the Republic of Singapore (Khemlani 2005).

3.2 Mapping of wall entities

In IFC 2x2, wall entities are exchanged in STEP as instances of *IfcWall* or *IfcWallStandardCase*. The *IfcWallStandardCase* entity type supports standard walls, which have the following characteristics: uniform overall thickness along the wall path; wall path follows a straight line or a circular arc; various offsets from the wall path may exist; single or multiple material layers and a constant or a varying height along the wall path. All other cases are considered non-standard and categorised as specific wall type, which is supported by the *IfcWall* entity.

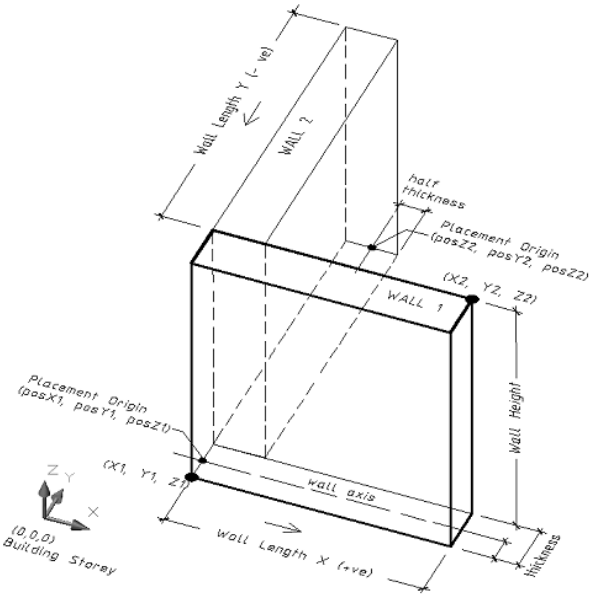


FIG. 4: An IfcWallStandardCase representation

The scope of *IfcWallStandardCase* and *IfcWall* entity exchange to FDS has been restricted to the following:

- Uniform thickness along the wall path;

- Path follows a straight line with no offsets and parallel to either the X or Y axis, hence orthogonal with respect to each other;
- Consists of a single uniform material layer;
- Have a constant height throughout the building storey;
- Vertical axis is perpendicular to the floor and ceiling planes;
- Base is at a constant elevation throughout the building storey.

Both standard and specific wall types have at least two shape representations in the IFC model, namely the wall axis and the wall body. Standard walls are typically defined by the centreline axis and the sectional profile. The wall axis provides a reference for the material layer offset and also defines the wall path. It is also used as a reference axis for the placement of walls and wall openings. As shown in Fig. 4, the placement origins of the walls are expressed in the coordinate system of the building storey.

The wall body may be represented by one of three possible types of solid model, as given by the *IfcShapeRepresentation* entity:

- Clipping representation is the geometric representation as the result of a Boolean subtraction on the CSG solid models. Wall elements are often clipped by the roof entity or other clipping planes.
- Swept Solid representation is the geometric representation of solid models created with profile sweeping using either linear extrusion or revolution techniques.
- Boundary Representation (BREP) describes a solid that is defined by its boundaries rather than by profiles and extrusions. In the IFC model, BREP geometry is the fallback position for any geometry than cannot reasonably be represented using parametric solids.

The wall representation type exchanged depends on the modelling methods used as well as on the IFC implementation in the BIM authoring application. The following observations have been made based on the implementation of Revit Building 9:

1. Walls having no 'constraints' (i.e. detached from the floor and the ceiling) are always exchanged as *IfcWallStandardCase* regardless of whether or not they contain any doors, windows or wall openings. The wall body in this case is always represented as *Swept Solid* type.
2. When a wall has either a bottom or top constraint, there are two possibilities, as follows:
    - If it contains a door or a wall opening, it is exchanged as *IfcWall* with BREP geometry.
    - If it is a solid wall without any openings, it is exchanged as *IfcWallStandardCase* and with *Clipping* representation type.

TABLE 1: Wall representation type exchanged by Revit Building 9

Case	**Top or bottom constraints**	**Opening type**	**Wall entity type**	**Wall body type**
1	No	Any opening	*IfcWallStandardCase*	*Swept Solid*
2	Yes	No openings	*IfcWallStandardCase*	*Clipping*
3	Yes	Door, window or wall opening	*IfcWall*	*BREP*

For Case 1 and Case 2 of the wall representation (Table 1), the height of the wall can be obtained from the depth attribute of the *IfcExtrudedAreaSolid* entity. When the wall body is represented by BREP geometry two approaches can be used to determine the height of the wall since the mapping of BREP geometry is not currently supported by the parser. One approach is to navigate the STEP file hierarchy down to the *IfcPolyLoop* of the *IfcFacetedBrep* entity, and determine the maximum Z coordinate of all the *IfcCartesianPoint* entities. A second simpler approach is based on the fact that the parser has the limitation that walls have a constant height within the same *IfcBuildingStorey*. So long as at least one wall in the model is represented by *IfcWallStandardCase*., the height of the wall can be obtained as per the methods given in Case 1 or Case 2 above and this is then taken to be the standard wall height applied to all the walls on the same building storey. This approach cannot be used where all walls in the model are represented by *IfcWall* entities.

The length of the wall for both *IfcWallStandardCase* and *IfcWall* can be obtained by navigating into the wall axis shape representation entity and the length on the wall centrelines has been adopted as the reference axis. Based on this convention, the walls can be positioned simply by using the given length and the thickness regardless of whether they have mitred or butt joints at corners (Fig. 5). The wall thickness can be obtained directly from the *IfcMaterialLayer* entity. Vector lengths of the wall are calculated by multiplying the length quantity by the wall direction. Walls are orthogonal with respect to each other and they can only be parallel to either X or Y axis as limited by the scope of the current mapping.

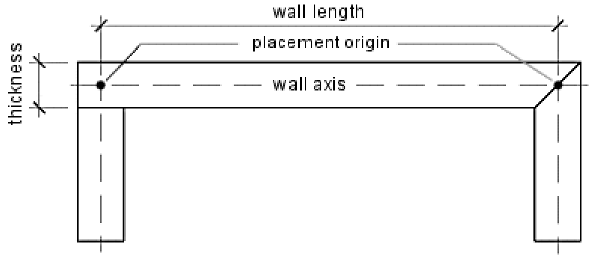


FIG. 5: Wall length and placement origin on the wall axis

The placement origin or the positional coordinates of the wall are assumed to be on the wall centreline as this is the method employed by Revit's IFC exchange convention. This assumption may not always be true, as some other BIM authoring applications may have the default placement origin set on the outer or inner face of the wall rather than the centreline.

To determine the sextuplet coordinates required to define a wall as a solid obstruction block in FDS, the mapping methodology shown in Figure 6 has been used where the parameters 'length_x' and 'length_y' refer to the vector lengths of the wall and 'thk/2' represents half the wall thickness, which is the amount of offset from the centreline to the face of the wall.

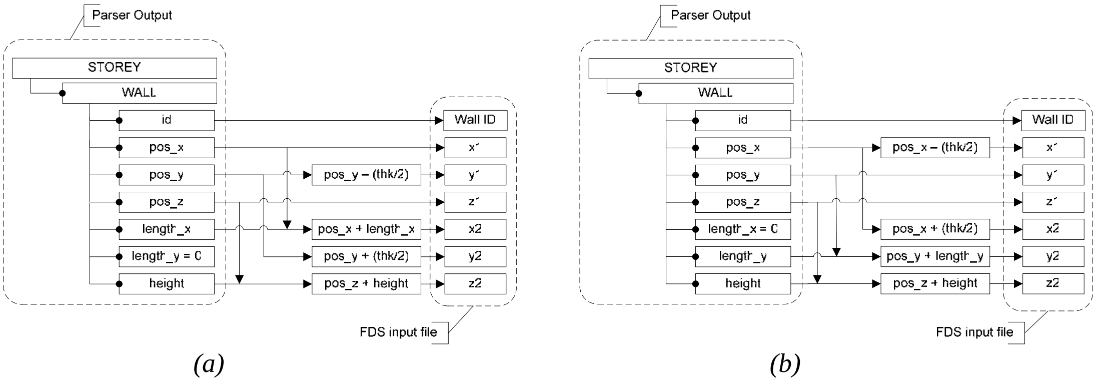


FIG. 6: Mapping diagram for walls parallel to the X and Y axes

3.3 Mapping IFC door and window openings

Door and window openings are exchanged as *IfcOpeningElement*. The relationships between the openings and the voids in the wall or the openings and the filling elements such as a door or a window are described by *IfcRelVoidsElement* or *IfcRelFillsElement* entities, respectively. Given an *IfcOpeningElement*, the associated wall can be determined from the *IfcRelVoidsElement*. Similarly, the door or window associated with an *IfcOpeningElement* can be determined from the *IfcRelFillsElement*, if it exists.

The local placement of door and window openings is exchanged as the horizontal and vertical offset distances from the placement origin of the wall to the edge of the opening. This can be obtained from the STEP file where the horizontal offset distance (in X-axis direction) is found from the placement origin of the wall to either the near or far edge of the opening depending on the direction of the opening (Fig. 7).

In accordance with the adopted convention, the placement origin or the positional coordinates of door and window openings must be on the wall centreline of the reference axis. The horizontal offset distance is taken to either the near or far edge of the opening depending on the direction of the opening relative to that of the parent wall. Where the wall opening is in the same direction as the parent wall, the offset distance is to the near edge of the opening relative to the local placement of the parent wall. Alternatively, where the wall opening is in the opposite direction to the parent wall, the offset distance is to the far edge of the opening relative to the local placement of the parent wall.

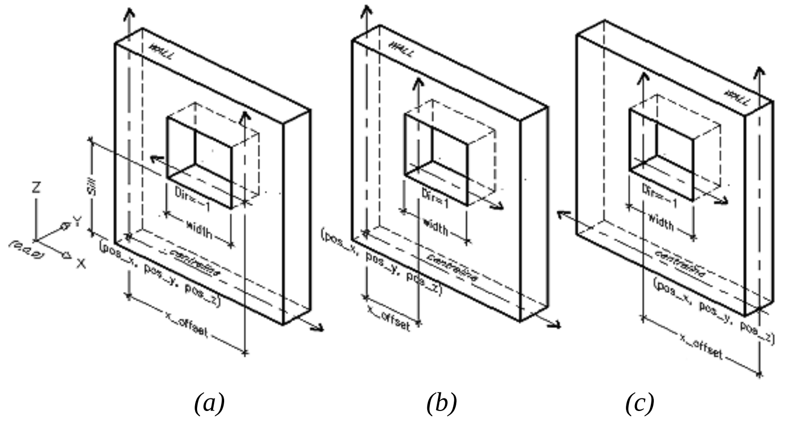


FIG. 7: Placement of door and window openings (a) wall in positive direction and opening in negative direction; (b) wall and opening both in positive direction; (c) wall in negative direction and opening in positive direction

It is convenient to allow the wall openings to inherit the direction of the parent wall. This is achieved by multiplying the width of the opening by the directions of the parent wall, which results in two vector components. As the walls are orthogonal with respect to each other and restricted to those parallel to either X or Y axis, only one of these vector components can represent the actual vector length (or width) of the opening and the other one would be zero.

Door swing direction can either be inward or outward. Similarly, window placement can either be on one side or the other relative to the parent wall. As FDS is generally interested only with the actual door opening, the door swing direction is of no particular relevance. Therefore, both windows and doors have exactly the same cases of placement directions relative to the parent wall, i.e. it is either positive or negative relative to the parent wall.

To determine the sextuplet coordinates required to define a wall opening as a void in FDS, the mapping methodology shown in Fig. 8 has been used where ‘width_x’ and ‘width_y’ are the vector widths of the opening and ‘thk/2’ represents half the wall thickness, which is the amount of offset from the centreline to the face of the wall. This offset includes an arbitrary 1.5 factor to provide a 25% additional wall thickness on each side of the centreline to ensure that the hole punches through the entire thickness of the FDS obstruction block.

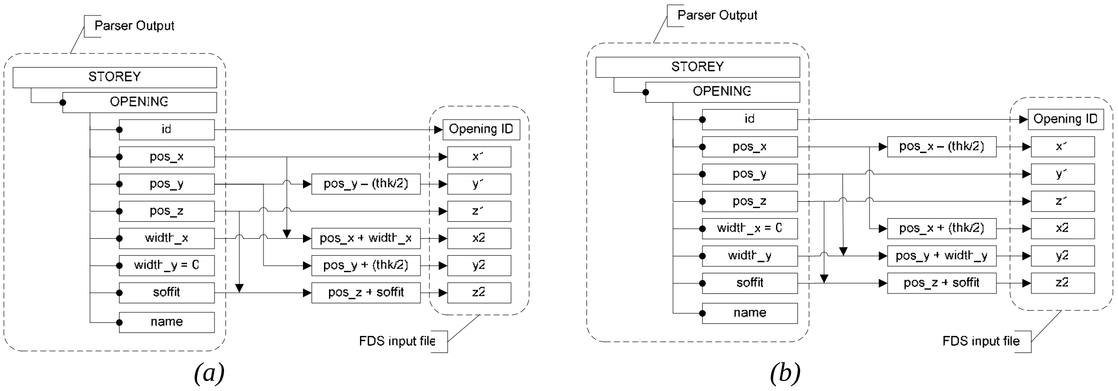


FIG. 8: Mapping diagram for openings (a) on walls parallel to the X axis; (b) on walls parallel to the Y axis

3.4 Mapping IFC wall openings

A wall opening is a void or hole through the wall without an associated door or a window entity. Wall opening objects must be treated differently as they differ in details to doors and windows openings and are exchanged differently in the Revit Building 9 STEP file.

As described in wall representation Case 3, when a wall opening is on a wall that is attached to the floor (i.e. with base constraints), the wall is exchanged as *IfcWall* with BREP representation. In this case, the opening would be part of the BREP profile of the wall geometry and would not be exchanged as *IfcOpeningElement* entity. Due to the current limitations of the parser, in order for the wall opening to be correctly exchanged as *IfcOpeningElement*, the base constraints of the wall must be removed in the source Revit Building model.

The height and width, and the local placement of the wall opening can be obtained from the *IfcAxis2Placement3D* entity. Unlike the door and window openings, the local placement of the wall openings is at the centre of the opening instead of at the edge (Fig. 9b). In accordance with the adopted convention, the final placement of the opening needs to be shifted from the centre to the edge closest to the placement origin of the wall (Fig. 9a). This is achieved by subtracting half of the width of the opening from the X axis offset, and subtracting half of the height of the opening from the Z axis offset.

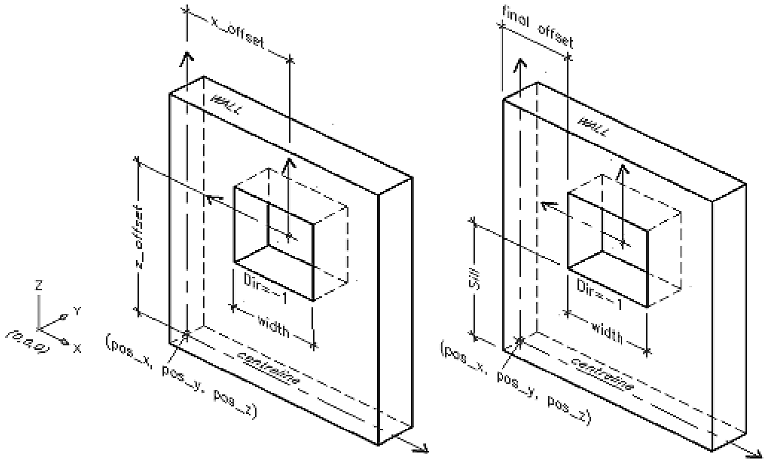


FIG. 9: Placement of wall openings (a) Revit Building model IFC representation; (b) final opening placement

3.5 FDS conversion application

The FDS conversion application uses HTML with embedded PHP code. Each function of the application is handled by a separate module that is server-side included on demand (Fig. 10). The FDS conversion application uses several external libraries that are all available as part of the PHP implementation on an Apache server, such as the Expat library to parse XML documents and the GD library to dynamically generate graphics.

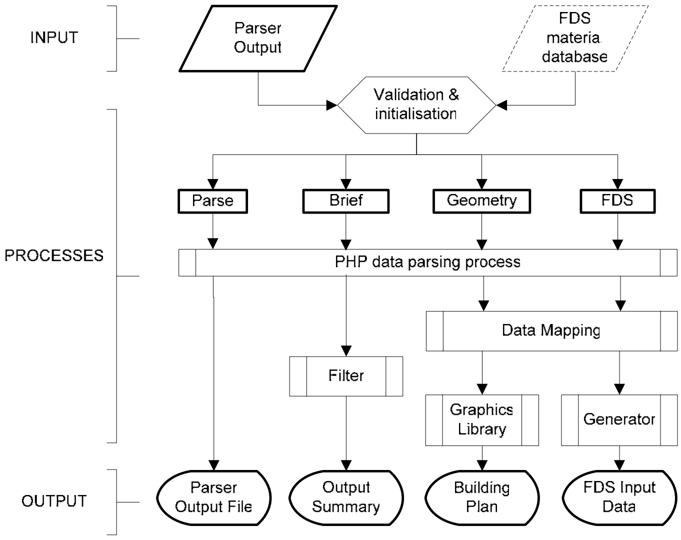
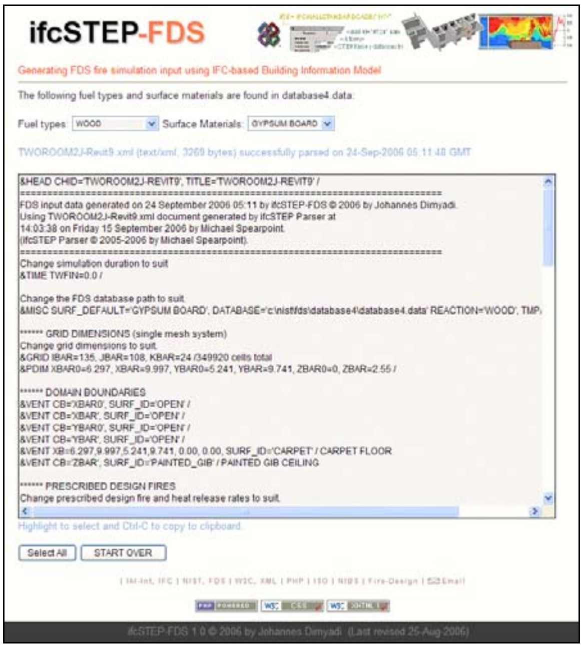


FIG. 10: Application flow diagram

The only input required by the FDS conversion application is the fire engineering subset of IFC model entities generated by the parser. The user may optionally specify the material database file installed with the FDS application on the local computer. The types of gas and solid phase materials defined in the database are retrieved and displayed for user's convenience and useful reference (Fig. 11a). The 'Geometry' feature generates a building plan geometry for display on-screen along with the mapping results for the walls and wall openings

(Fig. 11b). This feature is useful as a preview to visually verify the mapping of building walls and wall openings. The ‘FDS’ feature displays on-screen the generated FDS input data (Fig. 11a). The displayed output text can be selected and copied to the clip-board for transfer to a text file.



ifcSTEP-FDS

Generating FDS fire simulation input using IFC-based Building Information Model

The following fuel types and surface materials are found in database4 data:

Fuel types: WOOD Surface Materials: GYPSUM BOARD

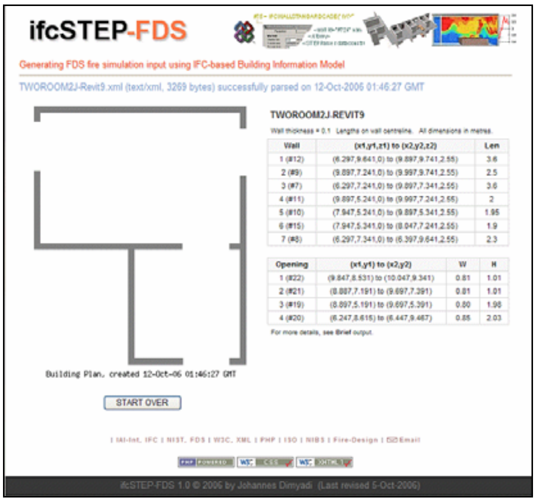
TWOROOM2J-Revit9.xml (text/xml, 3269 bytes) successfully parsed on 24-Sep-2006 05:11:48 GMT

```
&HEAD CHID='TWOROOM2J-REVIT9', TITLE='TWOROOM2J-REVIT9' /
==========================================================================
FDS input data generated on 24 September 2006 05:11 by ifcSTEP-FDS © 2006 by Johannes Dimyadi.
Using TWOROOM2J-Revit9.xml document generated by ifcSTEP Parser at
14:03:38 on Friday 15 September 2006 by Michael Spearpoint.
(ifcSTEP Parser © 2005-2006 by Michael Spearpoint).
==========================================================================
Change simulation duration to suit
&TIME TWFIN=0.0 /

Change the FDS database path to suit
&MISC SURF_DEFAULT='GYPSUM BOARD', DATABASE='c:\nistfds\database4\database4.data' REACTION='WOOD', TMP

****** GRID DIMENSIONS (single mesh system)
Change grid dimensions to suit
&GRID IBAR=135, JBAR=108, KBAR=24 /349920 cells total
&PDIM XBAR0=6.297, XBAR=9.997, YBAR0=5.241, YBAR=9.741, ZBAR0=0, ZBAR=2.55 /

****** DOMAIN BOUNDARIES
&VENT CB='XBAR0', SURF_ID='OPEN' /
&VENT CB='XBAR', SURF_ID='OPEN' /
&VENT CB='YBAR0', SURF_ID='OPEN' /
&VENT CB='YBAR', SURF_ID='OPEN' /
&VENT XB=6.297,9.997,5.241,9.741, 0.00, 0.00, SURF_ID='CARPET' / CARPET FLOOR
&VENT CB='ZBAR', SURF_ID='PAINTED_GIB' / PAINTED GIB CEILING

****** PRESCRIBED DESIGN FIRES
Change prescribed design fire and heat release rates to suit
```

Highlight to select and Ctrl-C to copy to clipboard

Select All
START OVER
| IAI-Int, IFC | NIST, FDS | W3C, XML | PHP | ISO | NIBS | Fire-Design | Email
ifcSTEP-FDS 1.0 © 2006 by Johannes Dimyadi (Last revised 25-Aug-2006)

(a)



ifcSTEP-FDS

Generating FDS fire simulation input using IFC-based Building Information Model

TWOROOM2J-Revit9.xml (text/xml, 3269 bytes) successfully parsed on 12-Oct-2006 01:46:27 GMT

TWOROOM2J-REVIT9

Wall thickness = 0.1 Lengths on wall centreline. All dimensions in metres.

Wall	(x1,y1,z1) to (x2,y2,z2)	Len
1 (#12)	(6.297,9.641,0) to (9.897,9.741,2.55)	3.6
2 (#9)	(9.897,7.241,0) to (9.997,9.741,2.55)	2.5
3 (#7)	(6.297,7.241,0) to (9.897,7.341,2.55)	3.6
4 (#11)	(9.897,5.241,0) to (9.997,7.241,2.55)	2
5 (#10)	(7.947,5.241,0) to (9.897,5.341,2.55)	1.95
6 (#15)	(7.947,5.341,0) to (8.047,7.241,2.55)	1.9
7 (#8)	(6.297,7.341,0) to (6.397,9.641,2.55)	2.3

Opening	(x1,y1) to (x2,y2)	W	H
1 (#22)	(9.847,8.531) to (10.047,9.341)	0.81	1.01
2 (#21)	(8.887,7.191) to (9.697,7.391)	0.81	1.01
3 (#19)	(8.897,5.191) to (9.697,5.391)	0.80	1.98
4 (#20)	(6.247,8.616) to (6.447,9.467)	0.85	2.03

For more details, see Brief output.

Building Plan, created 12-Oct-06 01:46:27 GMT

START OVER
| IAI-Int, IFC | NIST, FDS | W3C, XML | PHP | ISO | NIBS | Fire-Design | Email
ifcSTEP-FDS 1.0 © 2006 by Johannes Dimyadi (Last revised 5-Oct-2006)

(b)

FIG. 11: Screenshots of FDS conversion output (a) FDS input file; (b) geometry view

In the FDS conversion process the computational domain side boundary planes have been defined as ‘OPEN’, which effectively means that they are flow transparent to the outside of the computational domain except for the areas obstructed by the external walls. Obstruction holes are defined as always open although the user can manually edit the FDS input file to specify the time in seconds at which the area is to be opened or closed.

Apart from the wall obstructions and wall openings, there are also several other predefined parameters included in the generated FDS input data. There are two predefined fire growth descriptions included, namely ‘Fire_Steady’ and ‘Fire_Growing’. In both cases, the initial Heat Release Rates per Unit Area (HRRPUA) specified is 2000 kW/m^2 and the growing fire parameter is defined by a simple ramping function. Three types of surface material definition and properties are included. The bottom computational domain boundary plane has conveniently been defined as having ‘CARPET’ surface material. Similarly, the top boundary plane has been defined to have ‘PAINTED_GIB’ surface material. The ‘UPHOLSERTY’ material can be used to attach to fuel packages. The ‘CARPET’ and ‘UPHOLSERTY’ properties are taken from the FDS database, and the ‘PAINTED_GIB’ is based on Eurefic cone calorimeter tests (Wade 2003). These predefined parameters are included by default for convenience and can be edited to suit using a text editor prior to the fire simulation.

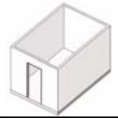
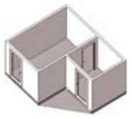
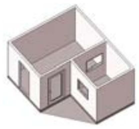
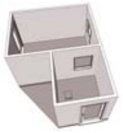
Once transferred and saved as an external file, the generated FDS input data can be immediately fed into FDS. As the default duration of the simulation is set to zero in the input file, the result of the simulation would initially be the SmokeView file containing the building geometry, which can be used to preview and verify the model. Alternatively the external file can be imported into a tool such as PyroSim for further editing before running a simulation.

3.6 Test case models

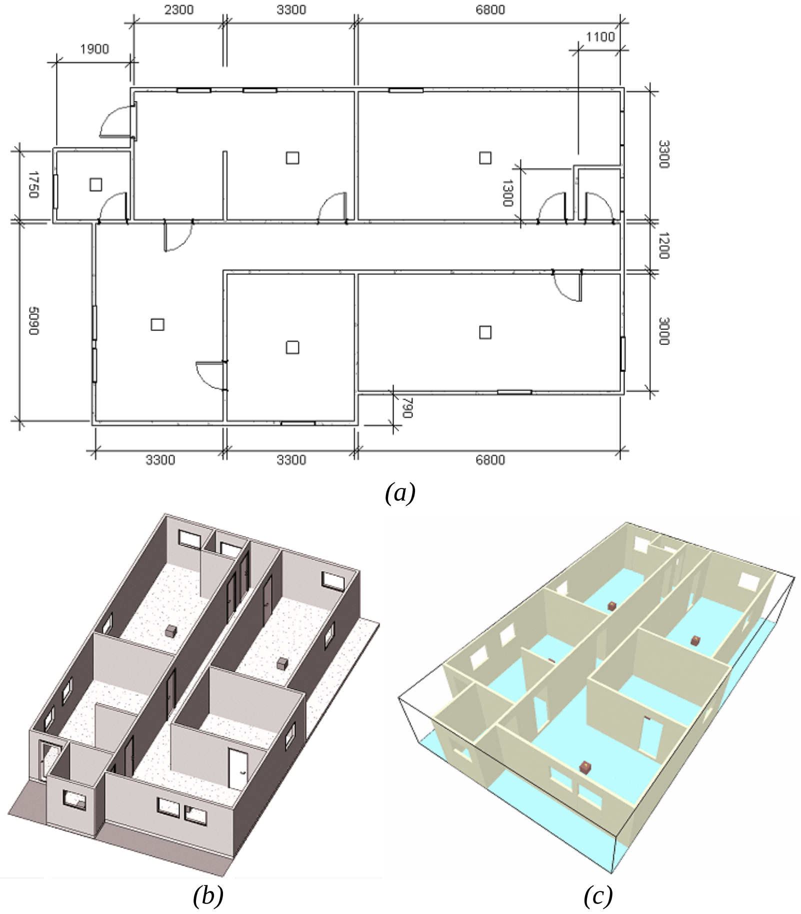
In order to examine the mapping implementation from Revit Building 9 to FDS through the conversion tool, a set of test case models were created as shown in Table 2. These test case models were used to verify the exchange of wall geometry as well as opening sizes and their positions from the original model to FDS. The IFC2x2 compliant STEP files for each of the test case models were obtained by exporting directly from the Revit Building 9 software package. The test case models were all constructed with the following common parameters and constraints:

1. A topographical surface entity was specifically created and included for exchange as *IfcSite* entity as required by the parser.
2. Due to the current limitations of the parser, all walls were constructed upright, equal height and orthogonal with respect to each other and were given a constant thickness throughout.
3. As *IfcSpace* entities are not required for the generation of objects in FDS, no space/room objects were specifically created or included in the models.
4. Door and window components included in Revit Building software package have been used. Wall openings have been created using the wall cut-out modelling tool.
5. All models consist of only one storey.

TABLE 2: Test case models

#	Name	Description	Revit 3D view
1	Single Room ISO9705 A	Single room 3.6 m x 2.4 m and 2.4 m high used in ISO 9705 fire test (ISO 1993) with a single 0.8 m by 2.0 m door. Walls are all attached to the floor slab. Used to verify the exchange of basic wall geometry.	
2	Single Room ISO9705 B	Same as ISO9705 A, except that the door has been replaced with a wall opening of the same size. Walls are all attached to the floor slab.	
3	Single Room 2	Single room 3.5 m x 2.3 m and 2.4 m high. Two doors with different swing directions on the external walls used to verify the exchange of doors with different ‘placement directions’.	
4	Single Room 3	Single room building with different types of opening, i.e. a window, a door and a plain opening, on an external wall. Used to observe how different types of opening are exchanged.	
5	Single Room 5A	Single room building with a window and a plain opening on one wall, a door and a window on another wall. Walls have no bottom constraints. Used to investigate the effect of setting bottom constraints and generation of a BREP profile for the wall.	
6	Single Room 5B	Same as Single Room 5A, except that walls are attached to the floor. Similar to Single Room 5A, to investigate the effect of setting bottom constraints and the generation of a BREP profile for the wall.	
7	Two-Room D	Two-Room building with two external doors and one internal door, different swing directions.	
8	Two-Room E	Two-Room with a plain opening on an external wall, a door on an external wall, a window on an external wall, and a window on an internal wall.	
9	Two-Room J	Two-Room with two external doors, one external window and one internal window.	

Finally a complex multi-room model with one external door, a series of internal doors and a series of windows on external walls was used to create a realistic modelling geometry that might be encountered in fire engineering practice. Walls had no top or bottom constraints and a box object was also placed in the centre of each room representing a piece of furniture or a group of furniture items that might be the source of a fire (Fig. 12). This geometry was successfully translated to FDS and viewed in SmokeView.



(a)

(b) (c)

FIG. 12: Multi-room model (a) floor plan; (b) Revit Building 9 3D view; (c) FDS output in Smokeview

3.7 Limitations and future work

The current work has a number of limitations, mainly in two areas that are somewhat interdependent. The parser currently only supports IFC 2x2 and is limited to the mapping of straight, orthogonal and equal height standard IFC wall and opening elements. Furniture items are currently exchanged only by their placement origin and no further details are exchanged. Certain basic IFC elements such as *IfcProject*, *IfcSite* and *IfcBuildingStorey* are prerequisites in the source model although they may not necessarily be created by default in a particular BIM tool.

The FDS conversion tool is only compatible to FDS version 4 and only supports single storey buildings. It also inherits the parser's constraints for mapping only straight and orthogonal elements. The computational grid resolution is preset for the average case and no optimisation algorithm has been implemented. The thermo-physical properties of materials and the design fire specifications are all user specified.

Suggested future work includes the support for IFC 2x3 and the soon to be released FDS version 5, mapping of diagonal and curved walls, support for multi-storey buildings, mapping of material properties and fire safety systems.

4. CONCLUSION

An FDS conversion tool has been developed in conjunction with a revised version of fire engineering IFC model parser. The tool successfully converted building geometries created in Revit Building 9 to generate the corresponding FDS input data. Given the complexity of the IFC data model and the variants in the BIM methods, the data mapping implementation is currently limited to basic building geometry.

While there has been a growing interest in the other domains in the AEC/FM industry to use the IFC data model, particularly for downstream applications, there is certainly a scope in the fire engineering domain that could benefit from this. The initiatives taken by the parser tool described here and subsequent interface applications would potentially provide a considerable cost saving and improvement on the quality of data exchanged for fire simulations and fire safety design in general.

FDS has widespread fire engineering applications and continues to be developed by its authors. It would be hoped that future releases of FDS might be able to utilise the IFC data model directly as a means of obtaining building geometry and properties. The work presented in this paper provides direction as to how this can be achieved.

5. ACKNOWLEDGEMENT

The fire engineering programme at the University of Canterbury is supported by the New Zealand Fire Service Commission.

6. REFERENCES

AIA Building Connections (2006). National Institute of Standards and Technology – data exchange standards activity. http://www.building-connections.info/organizations/nist.html.

Amor, R. (1997). Product models in design and engineering. Building Research Establishment, Garston, UK.

Cox, G., Kumar, S. (2002). Modeling enclosure fires using CFD. Section 3 / Chapter 8, The SFPE Handbook of Fire Engineering, 3rd edition (DiNenno ed.), NFPA, Quincy, MA.

Dimyadi, J. A. W. (2007). Generating FDS fire simulation input using IFC-based building information model. Masters of Engineering in Fire Engineering Report, University of Canterbury, New Zealand.

Forney G. P., McGrattan, K. B. (2006). User Guide for SmokeView Version 4 – A tool for visualizing Fire Dynamics Simulation data. NIST Special Publication 1017. National Institute of Standards and Technology, Gaithersburg, USA.

Frost, I., Patel, M. K., Galea, E. R., Rymacrzyk, P., Mawhinney, R. N. (2001). A semi-automated approach to CAD input into field based fire modelling tools. Proc. of the 9th International Fire Science and Engineering Conference, Interflam 2001, vol. 2, pp. 1421-1426, Edinburgh, Scotland.

IAI (2006). International Alliance for Interoperability. http://www.iai-international.org/About/History.html.

ISO 9705-33 (1993). Fire tests on building materials and structures - Part 33: Full-scale room test for surface products.

ISO 10303-21 (2002). Industrial automation systems - exchange of product model data - Part 21: Implementation methods; clear text encoding of the exchange structure.

Khemlani, L. (2005). CORENET e-PlanCheck: Singapore's automated code checking system. AECbytes "Building the Future". http://www.aecbytes.com/buildingthefuture/2005/CORENETePlanCheck.html.

Liebich, T. (2002). Standard analysis – current AEC situation – building models. Final report, ProdAEC project IST-2001-32035.

Liebich, T., Adachi, Y., Forester, J., Hyvarinen, J., Karstila, K., Wix, J. (2006). IFC 2x Edition 3. International Alliance for Interoperability. http://www.iai-international.org/Model/R2x3_final/index.htm.

McGrattan, K. B., editor (2005). Fire Dynamics Simulator (Version 4) technical reference guide. NIST Special Publication 1018. National Institute of Standards and Technology, Gaithersburg, USA.

McGrattan, K. B., Forney, G. P. (2005). Fire Dynamics Simulator (Version 4) user's guide. NIST Special Publication 1019. National Institute of Standards and Technology, Gaithersburg , USA.

NIST (2006). DXF2FDS documentation, NIST Fire Dynamics Simulator (FDS) and Smokeview. http://fire.nist.gov/fds/.

Spearpoint, M. J. (2007). Transfer of architectural data from the IFC model to a fire simulation software tool. Journal of Fire Protection Engineering , Vol. 17, No. 4, pp. 271-292.

Thompson, P. A., Marchant, E. W. (1995). A computer model for the evacuation of large building populations. Fire Safety Journal, Vol. 24, pp. 131-148.

Thunderhead Engineering Consultants (2006). PyroSim user manual (2006.2). Thunderhead Engineering Consultants in collaboration with The RJA Group Incorporated.

Wade, C. A. (2003). A user's guide to BRANZFIRE 2003. Building Research Association of New Zealand, Judgeford, Porirua City, New Zealand.