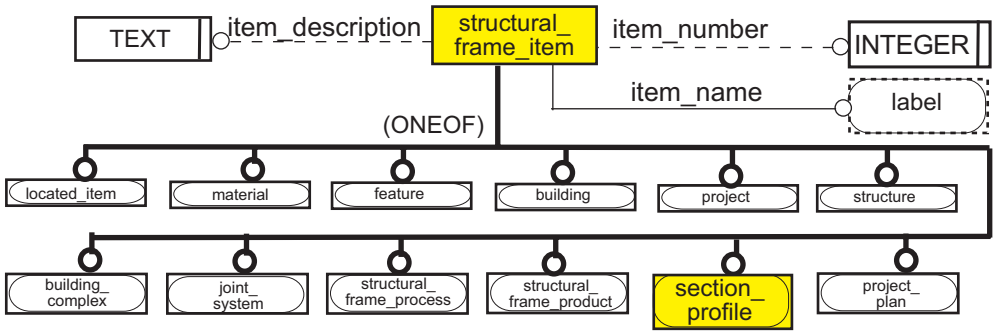
Section Profiles and Profile Libraries

C. Eastman

Each of the three different views of the CIS/2 model rely on common facilities to define structural steel physical members. Thus we have defined them here in one place. Physical members are basically defined as a section profile with end cuts and features that describe joints, holes and castellations, brackets and attachment points. This tutorial deals with section profiles.

There are two basic ways to define a section profile, *explicitly* by defining its section parameters, and *by reference* to a catalog or library. Most translators will need to support both methods. Both are described here.



FigureOne: The supertype of section_profile, structural_frame_item.

At the top level, section_profile is a subtype of structural_frame_item, as shown in Figure One. It inherits an item_name, item_number and optional item_description. As can be seen, structural_ frame_item is also the supertype of a number of other Entities.

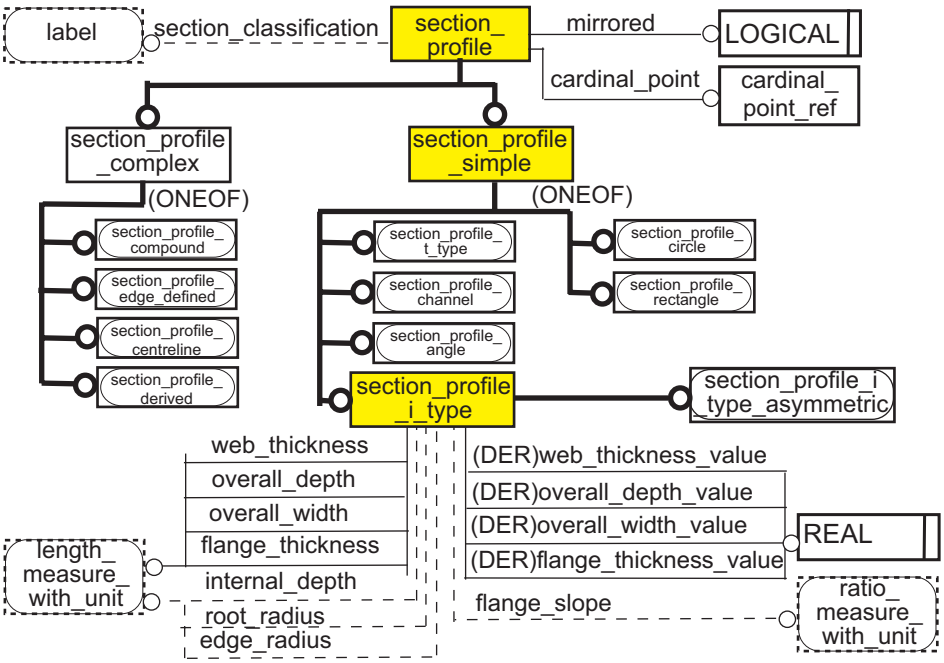


Figure Two: the definition of section_profile *and its simple subtype, the i_type section.*

Section_profile is shown in EXPRESS-G in Figure Two. It has three attributes:
a cardinal_number reference to a point, point_ref that is the base point for the section, a mirrored logical flag. Section_profile has two subtypes, simple and complex. The simple section is elaborated here and consist of standard fixed size cross sections. The simple section is defaulted to a symmetrical section; a subtype allows it to be asymmetrical (not defined here).

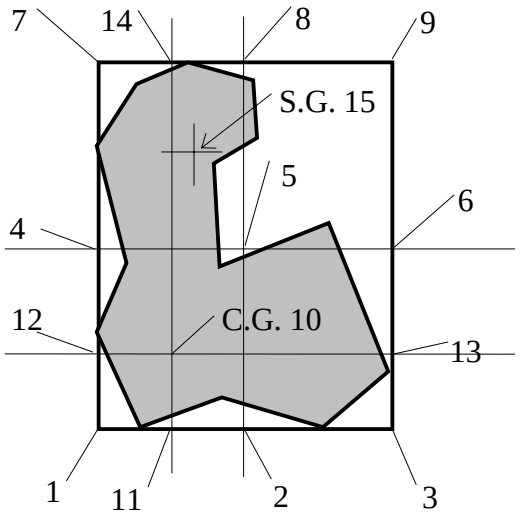
The corresponding EXPRESS code follows.

```
ENTITY structural_frame_item
SUPERTYPE OF (ONEOF
                (building,
                building_complex,
                feature,
                joint_system,
                located_item,
                material,
                project,
                project_plan,
                project_plan_item,
                section_profile,
                site,
                structural_frame_process,
                structural_frame_product,
                structure));
        item_number : INTEGER;
        item_name : label;
        item_description : OPTIONAL text;
END_ENTITY;

ENTITY section_profile
SUPERTYPE OF (ONEOF(section_profile_simple, section_profile_complex))
SUBTYPE OF (structural_frame_item);
        section_classification : OPTIONAL label;
        cardinal_point : cardinal_point_ref;
        mirrored : LOGICAL;
END_ENTITY;
```

Explicit section_profiles are defined by selecting the appropriate section_profile subclass and using the parameters provided to explicitly define the section parameters. In the long run, they should only be used for when an exchange does not have available a common reference to a catalog or library of sections.

```
ENTITY section_profile_complex
ABSTRACT SUPERTYPE OF (ONEOF
    (section_profile_compound,
    section_profile_edge_defined,
    section_profile_centreline,
    section_profile_derived))
SUBTYPE OF (section_profile);
END_ENTITY;

ENTITY section_profile_simple
ABSTRACT SUPERTYPE OF (ONEOF
    (section_profile_i_type,
    section_profile_t_type,
```

```
    section_profile_channel,
    section_profile_angle,
    section_profile_circle,
    section_profile_rectangle))
SUBTYPE OF (section_profile);
END_ENTITY;
```



1 - bottom left
2 - bottom center
3 - bottom right
4 - mid-depth left
5 - mid-depth center
6 - mid-depth right
7 - top left
8 - top center
9 - top right
10 - geometric centroid
11 - bottom in line with the geometric centroid
12 - left in line with the geometric centroid
13 - right in line with the geometric centroid

Figure Three: the reference points to be used in identifying the base point in cardinal_point_ref

A figure from the preliminary documentation of CIS/2, showing the alternative base points to be selected for a section_profile is shown in Figure Three. The number of cardinal references have been expanded to eighteen in the final documentation.

If we flatten the inheritances for any subtype of section_profile, we get:

```
ENTITY section_profile
structural_frame_item:
        (
          item_number : INTEGER;
          item_name : label;
          item_description : OPTIONAL text;
        );
          section_classification : OPTIONAL label;
          cardinal_point : cardinal_point_ref;
          mirrored : LOGICAL;
END_ENTITY;
```

So any type of section_profile carries the attributes described above.

We detail only the most common type of section. The section_profile_i_type is defined here with four required length measures: overall depth and width, flange thickness and web thickness. It also has four optional measures and four derived measures. Additional measures are required for asymmetrical sections. The EXPRESS entities for the section profile is shown below.

```
ENTITY section_profile_i_type
SUPERTYPE OF (section_profile_i_type_asymmetric)
```

```
SUBTYPE OF (section_profile_simple);
        overall_depth : positive_length_measure_with_unit;
        overall_width : positive_length_measure_with_unit;
        web_thickness : positive_length_measure_with_unit;
        flange_thickness : positive_length_measure_with_unit;
        internal_depth : OPTIONAL positive_length_measure_with_unit;
        flange_slope : OPTIONAL ratio_measure_with_unit;
        root_radius : OPTIONAL positive_length_measure_with_unit;
        edge_radius : OPTIONAL positive_length_measure_with_unit;
DERIVE
        overall_depth_value : REAL := overall_depth.value_component;
        overall_width_value : REAL := overall_width.value_component;
        web_thickness_value : REAL := web_thickness.value_component;
        flange_thickness_value : REAL := flange_thickness.value_component;
WHERE
        WRS13 : flange_thickness_value < (overall_depth_value/2);
        WRS14 : web_thickness_value < overall_width_value;
END_ENTITY;
```

Corresponding P21 file entires for a I-type section profiles are shown below.

```
#3648=SECTION_PROFILE_I_TYPE(102,'W16x36','ASTM spectification A6
1994',
'W flange',5,.F., #2141,#2143,#2145,#2147,$,$,$,$);
#2141=POSITIVE_LENGTH_MEASURE_WITH_UNIT(POSITIVE_LENGTH_MEASURE
(402.8439912796),#400);
#2143=POSITIVE_LENGTH_MEASURE_WITH_UNIT(POSITIVE_LENGTH_MEASURE
(177.4190033912),#400);
#2145=POSITIVE_LENGTH_MEASURE_WITH_UNIT(POSITIVE_LENGTH_MEASURE
(7.493000423908),#400);
#2147=POSITIVE_LENGTH_MEASURE_WITH_UNIT(POSITIVE_LENGTH_MEASURE
(10.9220001816),#400);
#2149=POSITIVE_LENGTH_MEASURE_WITH_UNIT(POSITIVE_LENGTH_MEASURE
(14.99999964237),
#400=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.METRE.));
```

In the above example the W16x36 member is defined, with its cardinal_point in the center of its cross section. It has 102 as its item_number. Its section is defined in millimeters. Such a section need only be defined once and referenced by multiple members with the same cross section. Keep in mind that the sectin carries the cardinal_point and if this changes, a different section_profile must be defined for each cardinal_point location.

The explicit definition of a member cross section should be written only when the appropriate reference library is not available. However, such cases may arise frequently. CIS/2 provides good means to reference items in a shared reference document. These reference items are loaded as data and accessible within a translator. The EXPRESS-G data structures for carrying them are shown in Figure Four.

The different reference sources are subtypes of item_ref_source. In most cases, these will be item_ref_source_standard. It has three attributes, giving a name, year of standard, standard organization and optionally a version. This typically will be defined with the reference data items. The actually data item_reference has two attributes, one to the source and one to the identifier within that reference.

The EXPRESS code for item_reference and its related entity types follows.

```
ENTITY item_reference
SUPERTYPE OF (ONEOF
                (item_reference_standard,
                item_reference_proprietary,
                item_reference_library));
        ref : identifier;
WHERE
        WRI7 : LENGTH(ref) > 0;
END_ENTITY;

ENTITY item_reference_assigned;
        assigned_reference : item_reference;
        assigned_to_item : structural_frame_item;
UNIQUE
        URI3 : assigned_reference, assigned_to_item;
END_ENTITY;
```

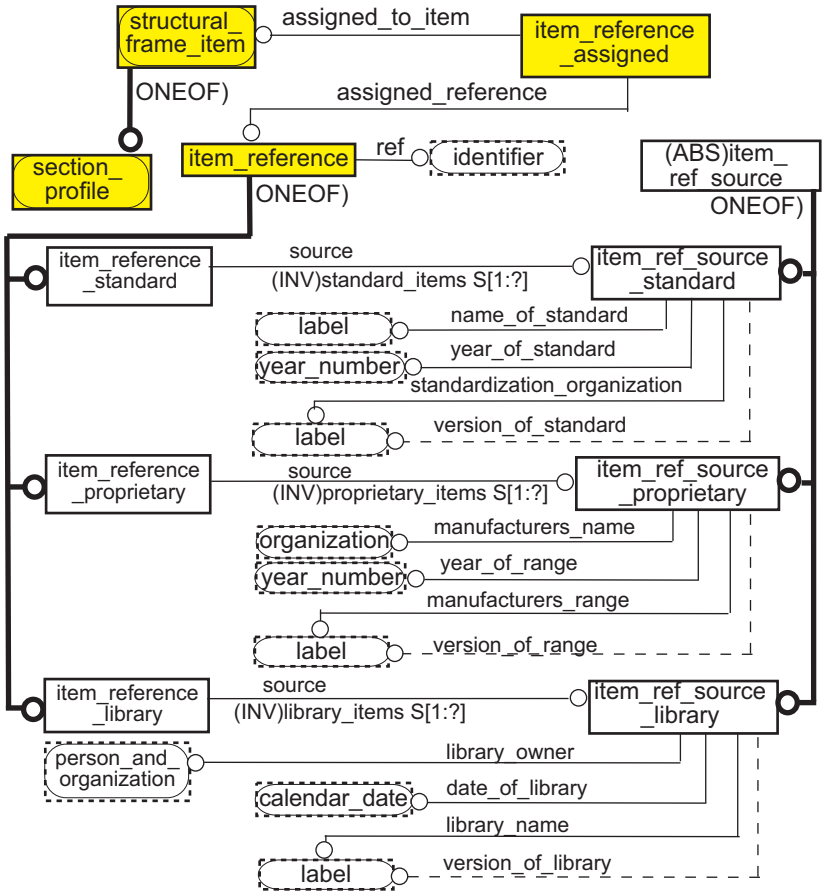


Figure Four: The definition of item reference for using shared references to define section profiles.

ENTITY item_reference_library

```
SUBTYPE OF (item_reference);
        source : item_ref_source_library;
UNIQUE
        URI4 : SELF\item_reference.ref, source;
END_ENTITY;

ENTITY item_reference_proprietary
SUBTYPE OF (item_reference);
        source : item_ref_source_proprietary;
UNIQUE
        URI5 : SELF\item_reference.ref, source;
END_ENTITY;

ENTITY item_reference_standard
SUBTYPE OF (item_reference);
        source : item_ref_source_standard;
UNIQUE
        URI6 : SELF\item_reference.ref, source;
END_ENTITY;

ENTITY item_ref_source
ABSTRACT SUPERTYPE OF (ONEOF
                (item_ref_source_standard,
                item_ref_source_proprietary,
                item_ref_source_library));
END_ENTITY;

ENTITY item_ref_source_library
SUBTYPE OF (item_ref_source);
        library_owner : person_and_organization;
        library_name : label;
        date_of_library : calendar_date;
        version_of_library : OPTIONAL label;

INVERSE
        library_items : SET [1:?] OF item_reference_library FOR source;
END_ENTITY;

ENTITY item_ref_source_proprietary
SUBTYPE OF (item_ref_source);
        manufacturers_name : organization;
        manufacturers_range : label;
        year_of_range : year_number;
        version_of_range : OPTIONAL label;
INVERSE
        proprietary_items : SET [1:?] OF item_reference_proprietary FOR source;
END_ENTITY;

ENTITY item_ref_source_standard
SUBTYPE OF (item_ref_source);
        standardization_organization : label;
        name_of_standard : label;
        year_of_standard : year_number;
        version_of_standard : OPTIONAL label;
INVERSE
        standard_items : SET [1:?] OF item_reference_standard FOR source;
```

END_ENTITY;

An item_reference may be a standard item, a proprietary_item or a library_item. The standard US section profiles are available in the *Shapes Database, U.S. Customary Units*, A.I.S.C.. An example might be:

```
#1106 = ITEM_REF_SOURCE_STANDARD('ASCI', 'ASTM_A6', 1994, $);
#1105 = ITEM_REFERENCE_STANDARD('W44X335', #1106);
#1107 = ITEM_REFERENCE_STANDARD('W44X290', #1106);
#1108 = ITEM_REFERENCE_STANDARD('W44X262', #1106);
#1109 = ITEM_REFERENCE_STANDARD('W44X230', #1106);
#1110 = ITEM_REFERENCE_STANDARD('W40X328', #1106);
```

These item_reference_source is the AISC handbook, ASTM A6 standard, 1994 edition. The item_reference_standards below it are from that standard and in the identifier field provide the identifiers to be used to reference items in the standard. A set of sections and references that assign sections follow.

```
#474=SECTION_PROFILE(10,$,$,5,.F.);
#475=SECTION_PROFILE(10,$,$,5,.F.);
#476=SECTION_PROFILE(10,$,$,5,.F.);
#477=SECTION_PROFILE(10,$,$,5,.F.);
#500=ITEM_REFERENCE_ASSIGNED(#1105,#474);
#501=ITEM_REFERENCE_ASSIGNED(#1107,#475);
#502=ITEM_REFERENCE_ASSIGNED(#1107,#476);
#503=ITEM_REFERENCE_ASSIGNED(#1109,#477);
```

There should be one item_reference_assigned for each section profile. No item_reference_assigned should reference a subtype of section_profile, as these are only used for explicit definitions of a section.

This walkthrough of both explicit and reference section profiles should get one started in exporting and importing sections from their application.