

二次元図面のSTEP/AP202 での実装に向けて

富士通STEP研究開発センター

取締役 井上 和

inouye@fqs.fujitsu.co.jp

STEPの規格体系

Application Protocol (and ATS)				
Common I 201 Draughting (W) I 202 Ass. Draught(C) I 203 Config Cont(C)	Lifecyvl Change W 208 Lifecycle (W)	Dimension Inspection X 219 Dim Inspection (X)	Ship W 215 Arrange (W) C 216 Mold (W) C 217 HVAC (W) C 218 構造 (W) W 226 Machinery (W)	Building C 225 Bld Elemnt (W) E 228 HVAC (O) W 230 Steel (W)
Mechanical Design C 204 Brep (C) C 205 Surface(W) X 206 WireFrame(X)	Composite Metal C 209 Composite (W) W 222 Structure (W)	Electro.Electric C 210 PCB design (W) W 211 PCB test(W) C 212 Electric (W) O 220 PCB manu (O)	Casting W 223 Cast (W) W 229 Forge (W)	PDM/TDP W 232 TDP (W)
Sheet Metal, Die E 207 Sheet M, Die(W)	NC, Machining E 213 NC Process (W) E 224 Feature(W)	Plant C 221 P&I (C) E 227 Arrange(W) W 231 Data (W)		Other O Opt data (O) O NATO (O)
	Automobile C 214 Automobile (W)			

Description
I 1 Overview I 11 EXPRESS X 12 EXPRESS-I O EXPRESS-X

O = Pre-Proposal A = Proposal -> NP W = Wdraft->CD C = CD->DIS E = DIS->FDIS F = FDIS->Int'l STD I = Publication X = Cancel
--

Integrated Resource	
Integrated Application Resource I 101 Draughting X 102 Ship hull X 103 Electro,elctric Connection C 104 FEA I 105 Kinematics W 106 Building Core Model	
Generic Integrated Resource I 41 Product I 42 Geometry I 43 Presen I 44 Config E 45 Material I 46 Presentation F 47 Tolerance X 48 Form feature E 49 Property	
AIC C 501 Edge C 502 Shell C 503 2DWF C 504 Annotation C 505 Structure C 506 Draught ele. C 507 Surface C 508 Non-Manifold C 509 Manifold C 510 WF C 511 Top-b C 512 Facet C 513 Brep C 514 Adv-Brep C 515 CSG X 516 Mech. design C 517 Mech. design (Geo) C 518 shadeing	

Implementation				
I 21 Phy.file	E 22 SDAI	C 23 C++	C 24 Late C	X 25 Fortran
C 26 IDL	O JAVA			

Conformance Test
I 31 General E 32 Test Envir X 33 ATS C 34 Part21Test W35 Part22 Test

STEP
STANDARD

AP202のSCOPE

- 機械及びAEC分野の製図交換の為の表現
- 製図管理情報
- 製図化された製品の版数情報
- 計算機が解釈可能な2又3次元製品形状モデル
- 図面ビューによる変換と見え方
- 寸法、表記又は塗りと製品幾何との連携
- 図面上の要素のグループピング

AP202のSCOPEに入らないもの

- 製図標準の作成
- 製図来歴の交換
- 図面に描かれない製品の幾何モデル
- 製品に関係しない図面
- 非平面の3次元表記
- シェーディング
- 製図形状モデルの表示
- 計算機が解釈可能でなければならないBOM
- 交差と幾何との連携
- 管理情報以外の表示出来ない属性

AP202の適合性クラス

- CC1: 製品形状のビューなしの表記
- CC2: 2次元(位相なし)の初等ワイヤースケーム
- CC3: 2次元(位相なし)のワイヤースケーム
- CC4: 2次元(位相付き)ワイヤースケーム
- CC5: 3次元(位相なし)のワイヤースケーム/サーフェス
- CC6: 3次元(位相付き)ワイヤースケーム
- CC7: ファセット境界表現
- CC8: 初等境界表現
- CC9: 高等境界表現
- CC10: 位相付きマニフォールドサーフェス

AP2020UOF

associative_dimension

draughting_shape_model

drawing_structure_and_administration

elements_of_annotation

elements_of_appearance

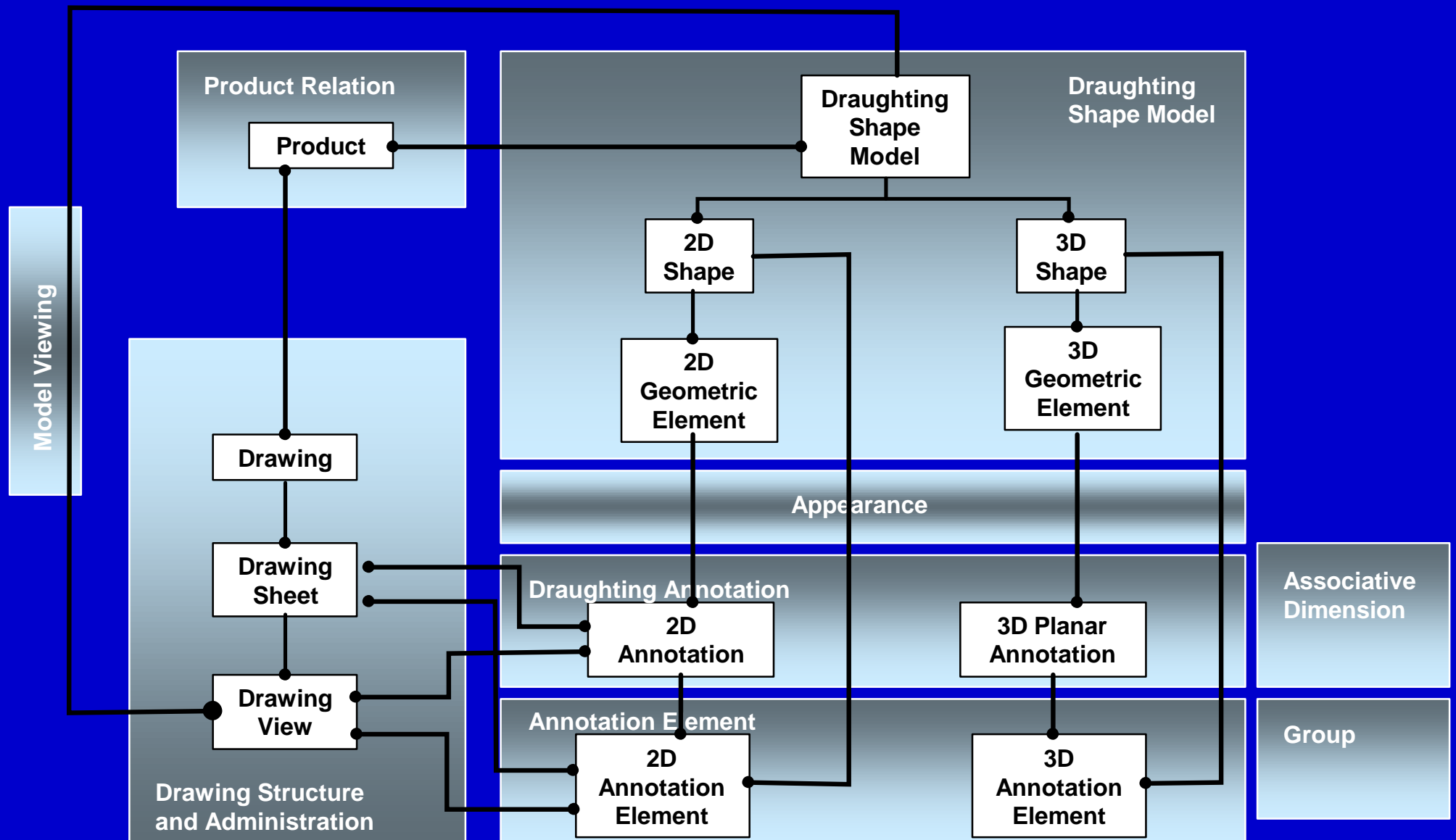
elements_of_draughting_annotation

grouping

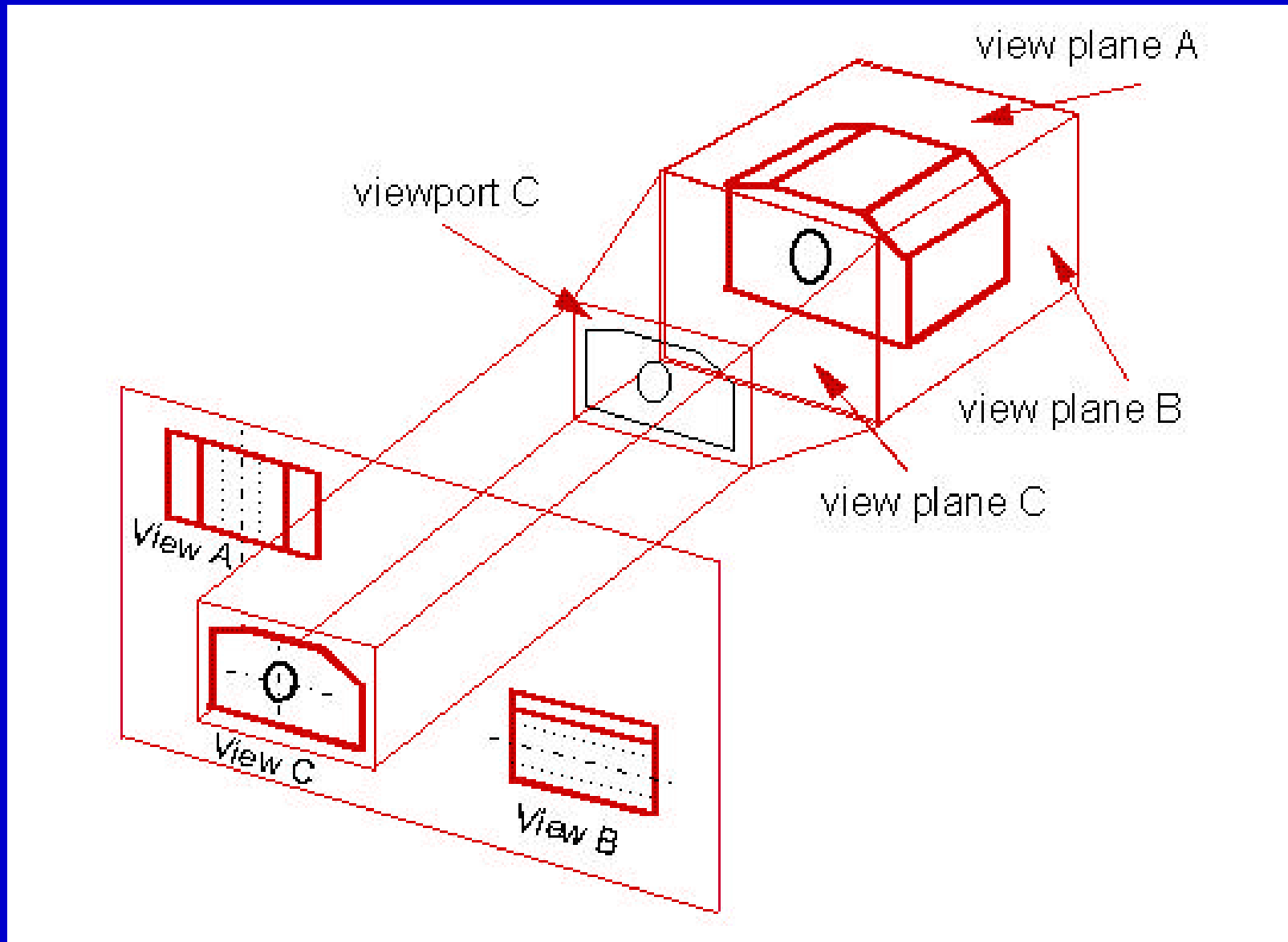
model_viewing

product_relation

UOF関連図

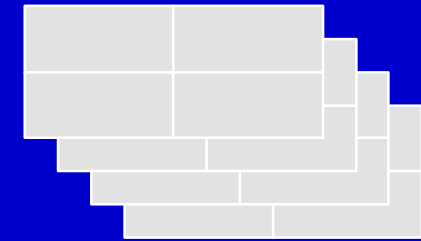
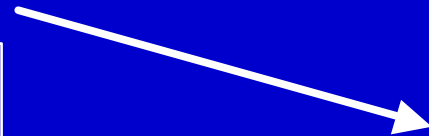
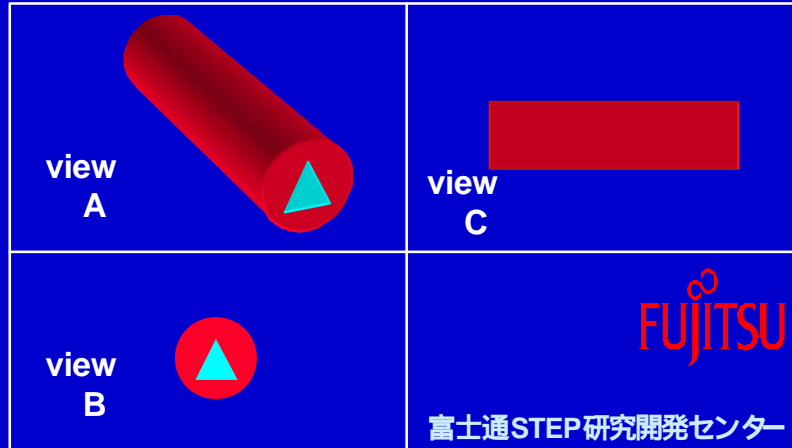


Model Viewing



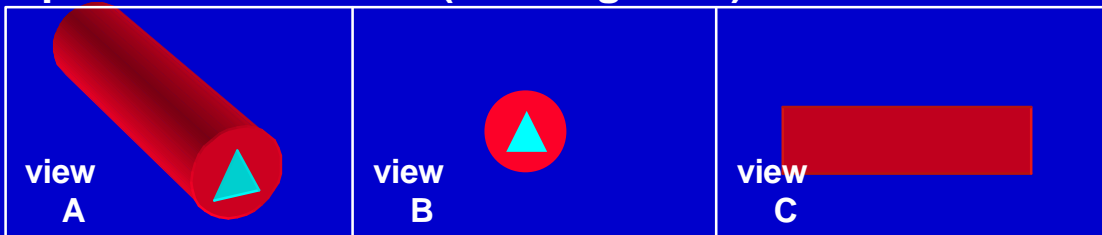
図面構造 (PDES Incの資料を流用)

presentation_area (drawing sheet)



drawing_revision (drawing)

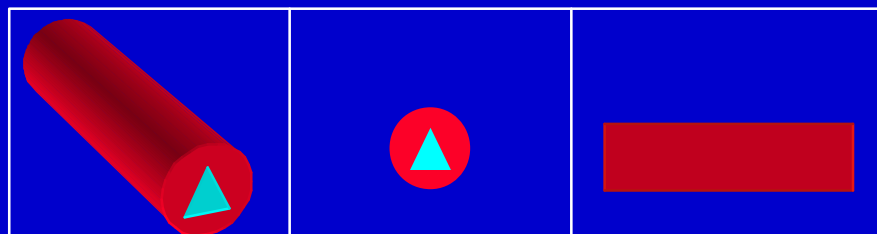
presentation_view (drawing view)



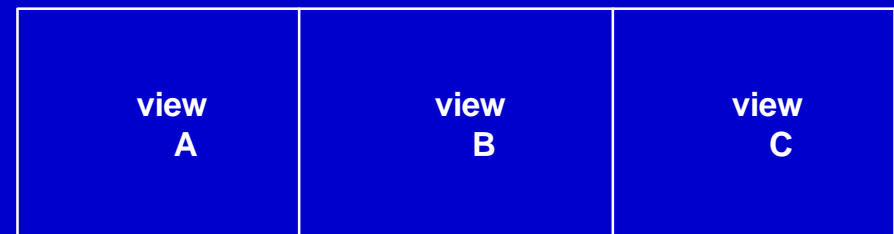
sheet_placed_annotation



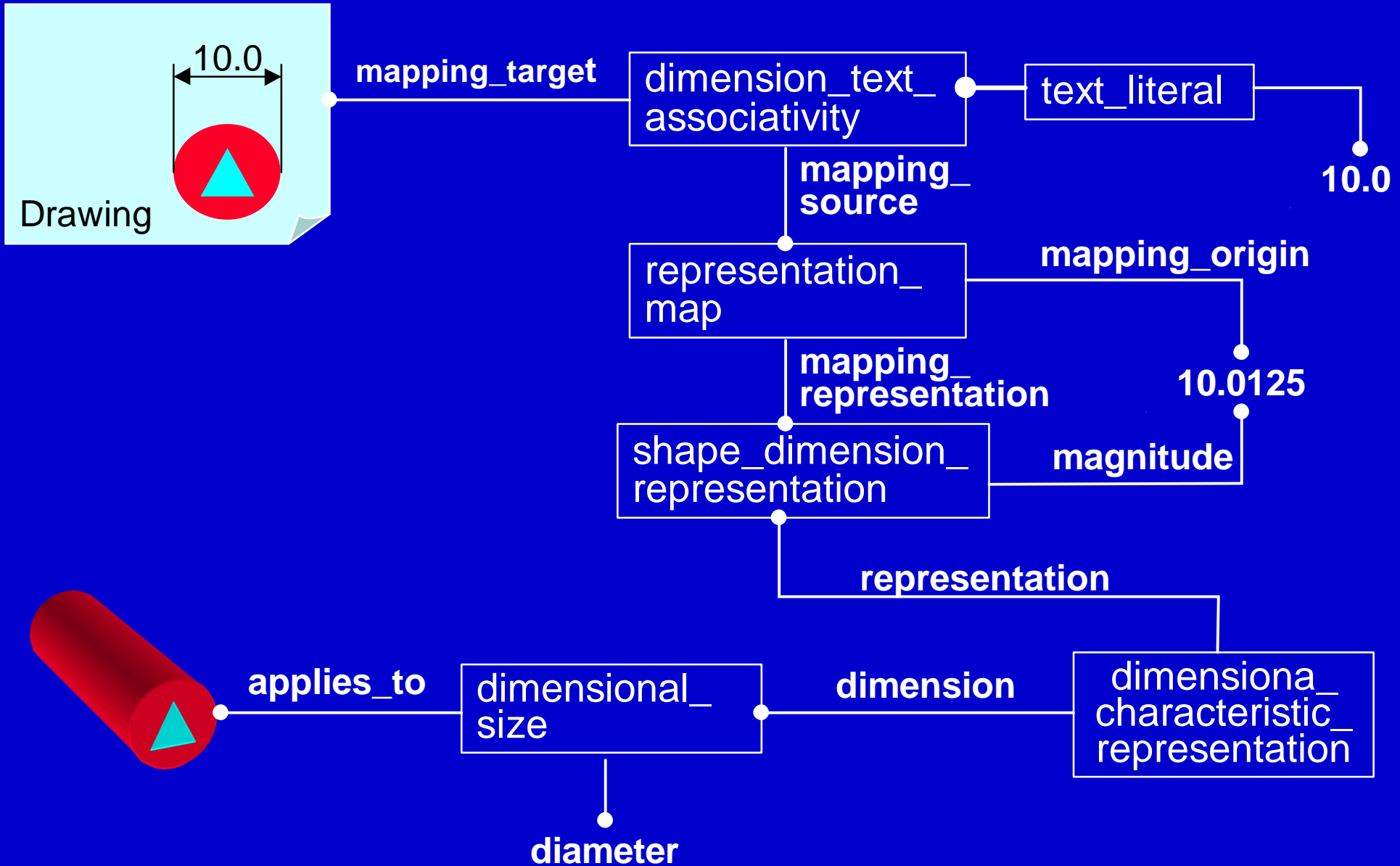
camera_model



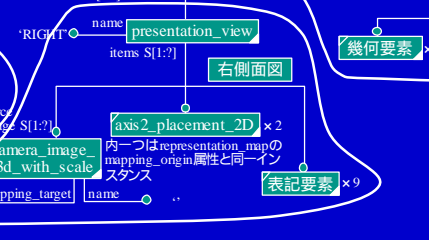
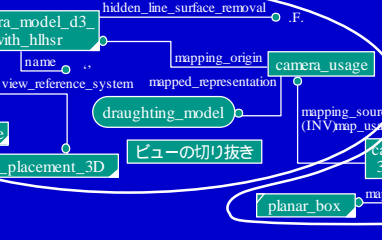
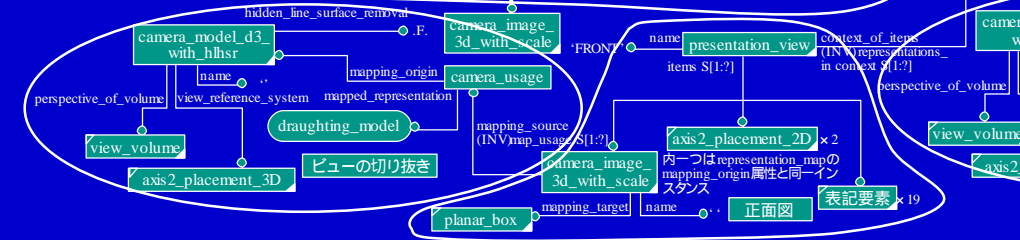
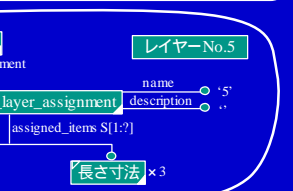
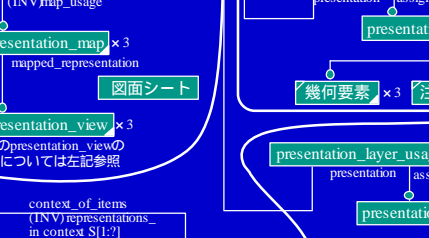
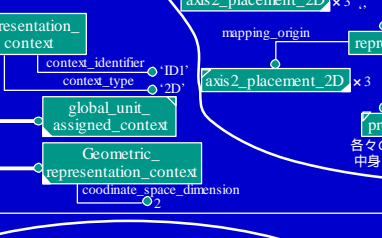
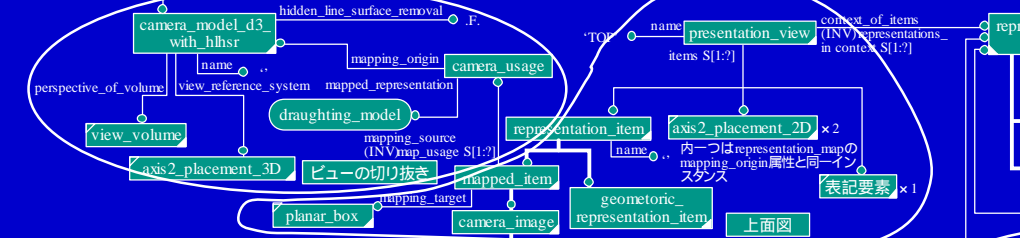
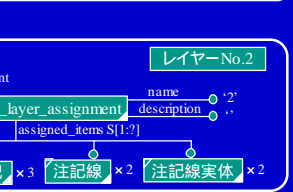
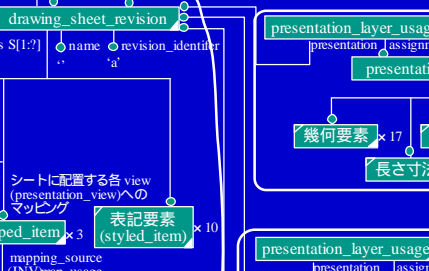
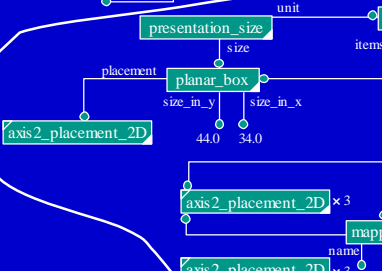
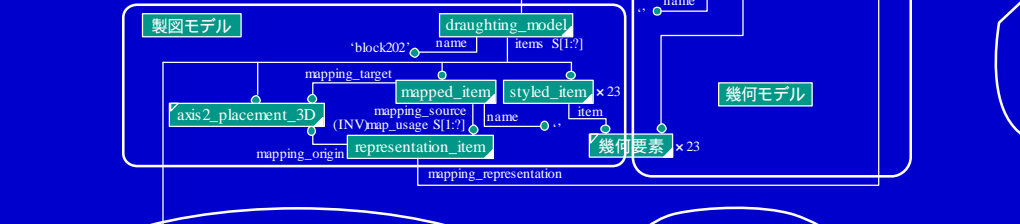
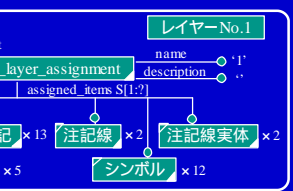
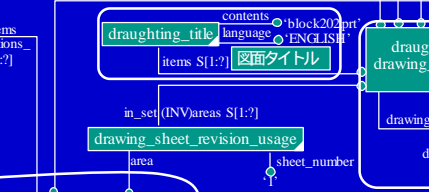
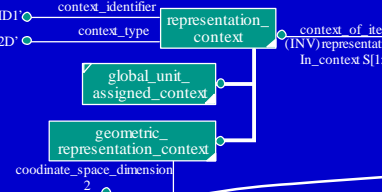
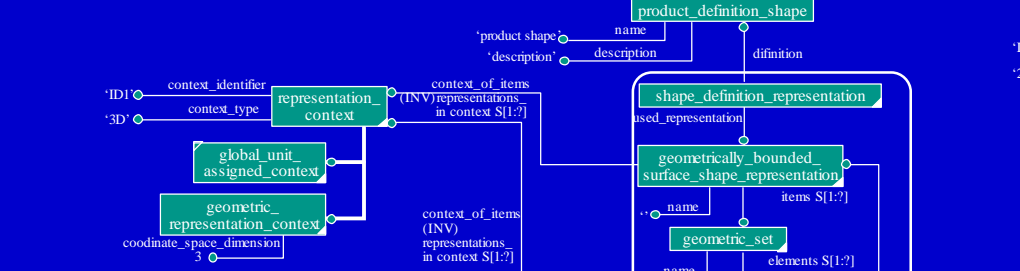
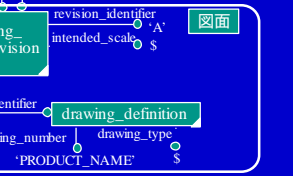
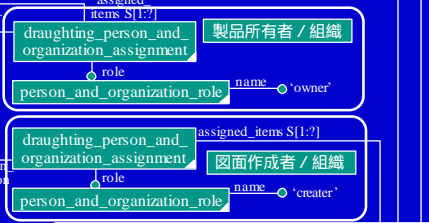
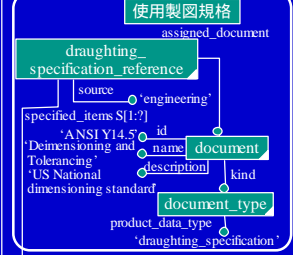
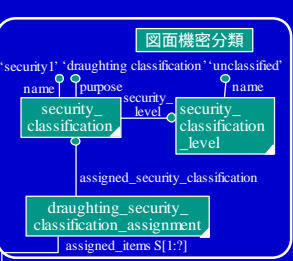
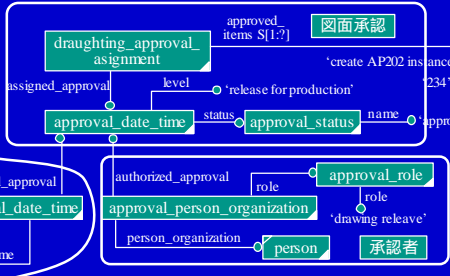
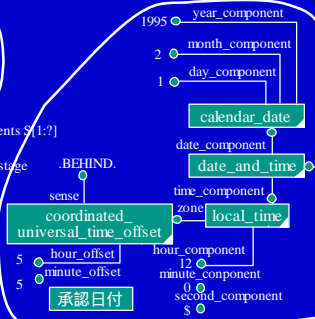
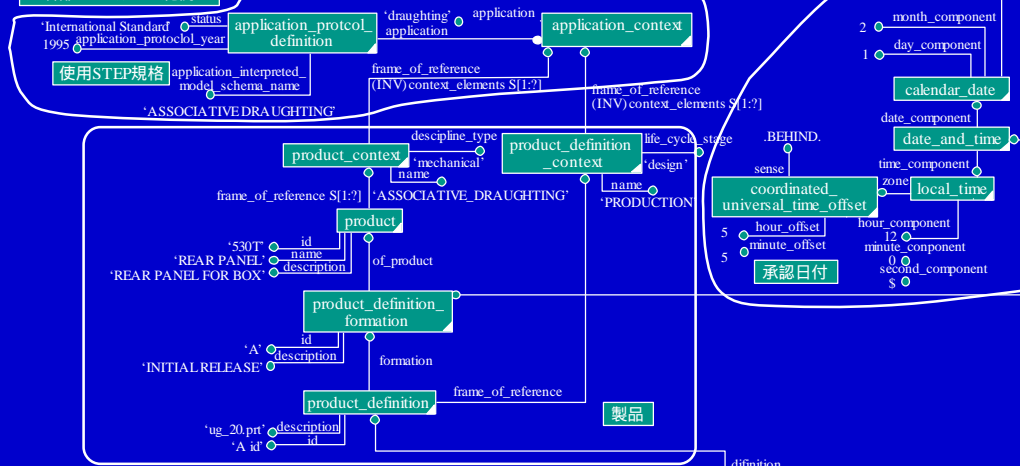
view_placed_annotation



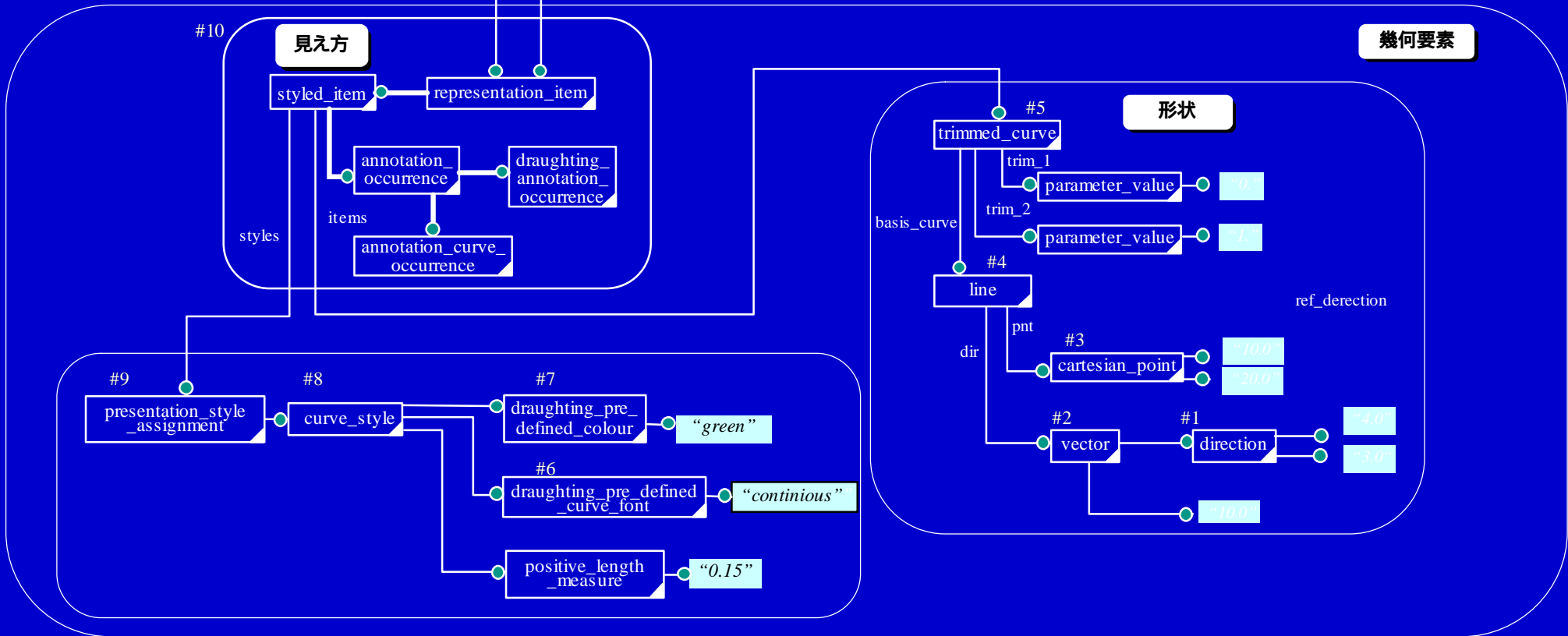
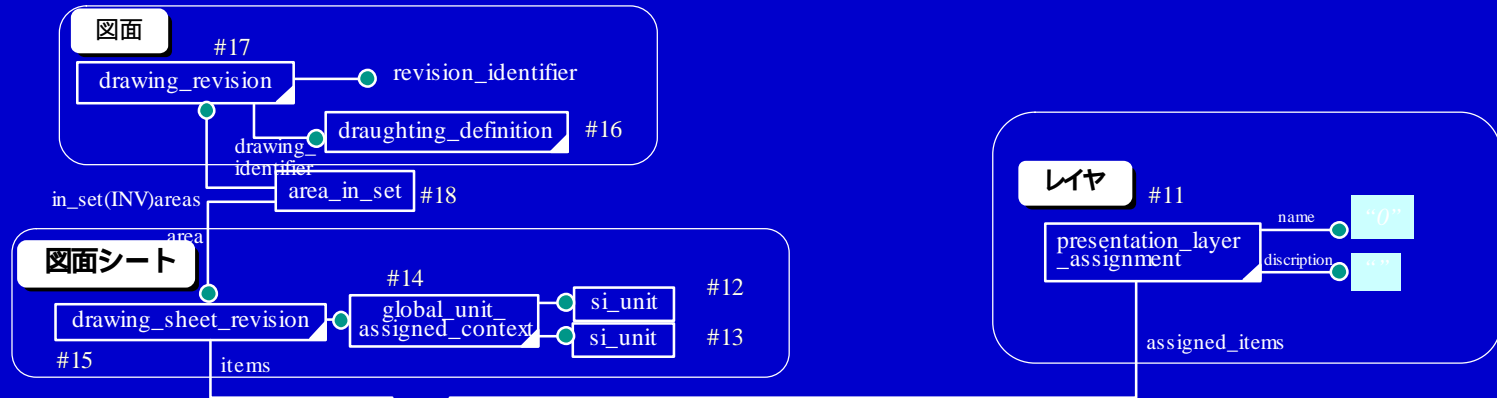
Associative Dimension



幹部インスタンス化例



線分インスタンス

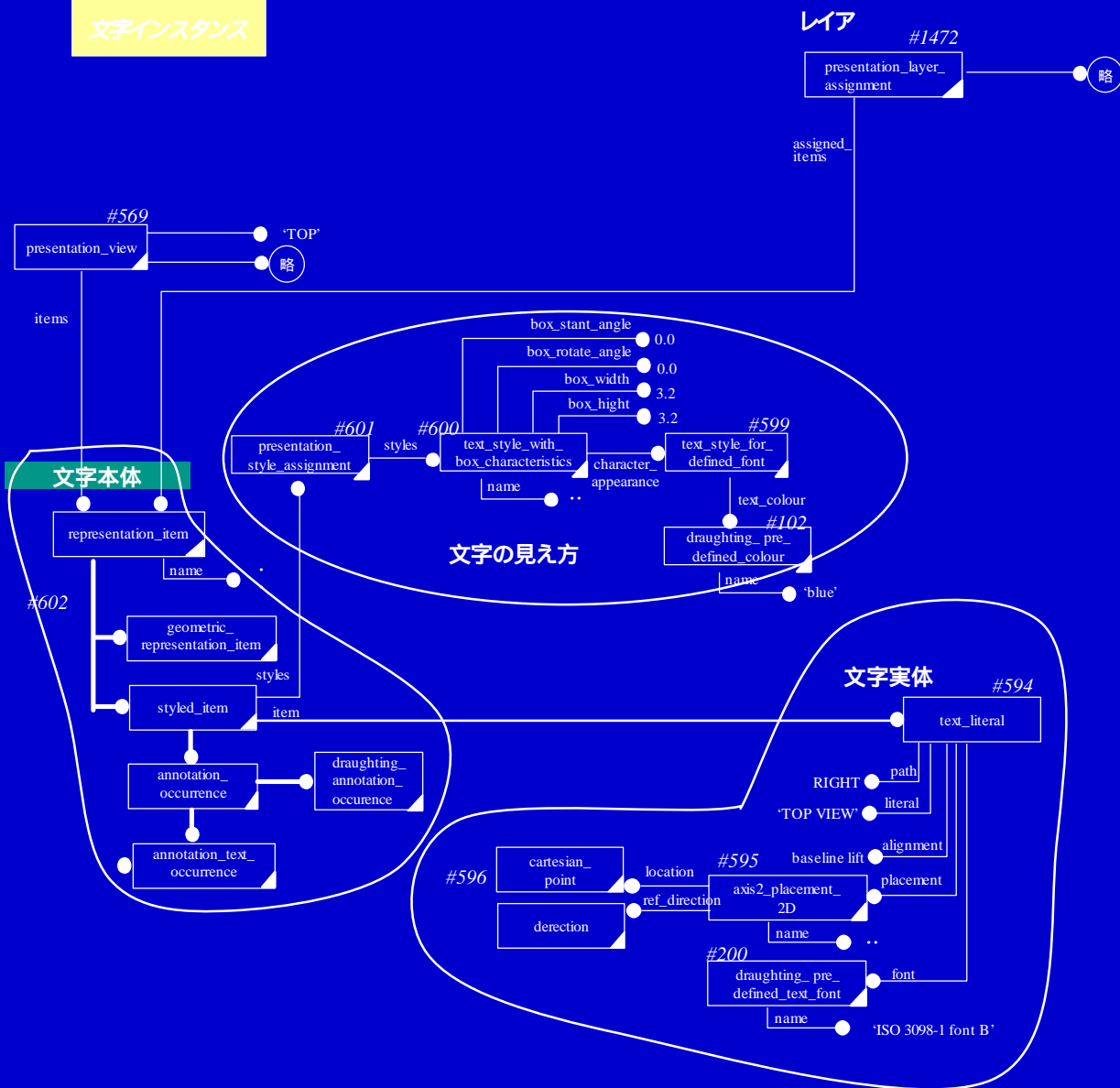


線分インスタンス(Part21ファイル)

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('ISO 10303-202'),'2;1');
FILE_NAME('Conceptual Part Example','1999-05-17 14:30:37',('<author>'),('FQS Ltd.'),
          '<preprocessor>', '<system>', '<authorisation>');
FILE_SCHEMA(('ASSOCIATIVE_DRAUGHTING'));
ENDSEC;
DATA;
#10 = DIRECTION('',(4.0,3.0));
#20 = VECTOR("#10,10.0);
#30 = CARTESIAN_POINT("(10.0,20.0));
#40 = LINE("#30,#20);
#50 = TRIMMED_CURVE("#40,(PARAMETER_VALUE(0.)), (PARAMETER_VALUE(1.)),.T.,.PARAMETER.);
#60 = DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#70 = DRAUGHTING_PRE_DEFINED_COLOUR('green');
#80 = CURVE_STYLE('<curvestylename>',#60,POSITIVE_LENGTH_MEASURE(0.15),#70);
#90 = PRESENTATION_STYLE_ASSIGNMENT((#80));
#100 = (ANNOTATION_CURVE_OCCURRENCE()
        ANNOTATION_OCCURRENCE()
        DRAUGHTING_ANNOTATION_OCCURRENCE()
        REPRESENTATION_ITEM()
        STYLED_ITEM((#90),#50));
#110 = PRESENTATION_LAYER_ASSIGNMENT('0',"(#100));
#120 = SI_UNIT(*,MILLI,METRE.);
#130 = SI_UNIT(*,$,RADIAN.);
#140 = GLOBAL_UNIT_ASSIGNED_CONTEXT('<context id>', '<context type>',(#120,#130));
#150 = DRAWING_SHEET_REVISION('<sheet name>',(#100),#140,'<sheet id>');
#160 = DRAWING_DEFINITION('<drawing definition id>',S);
#170 = DRAWING_REVISION('<drawing revision id>',#160,S);
#180 = AREA_IN_SET(#150,#170);
ENDSEC;
END-ISO-10303-21;
```

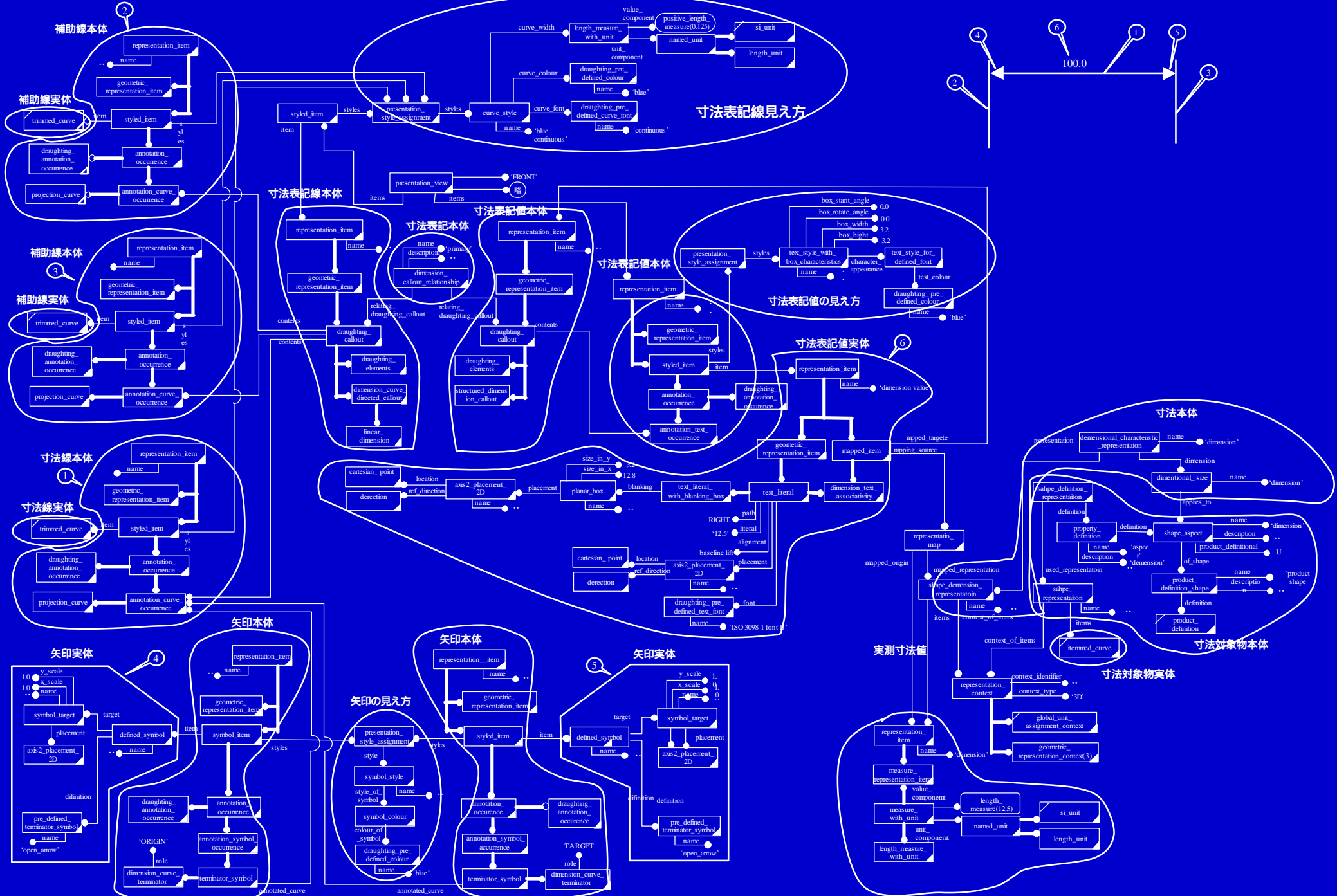
文字インスタンス

文字インスタンス

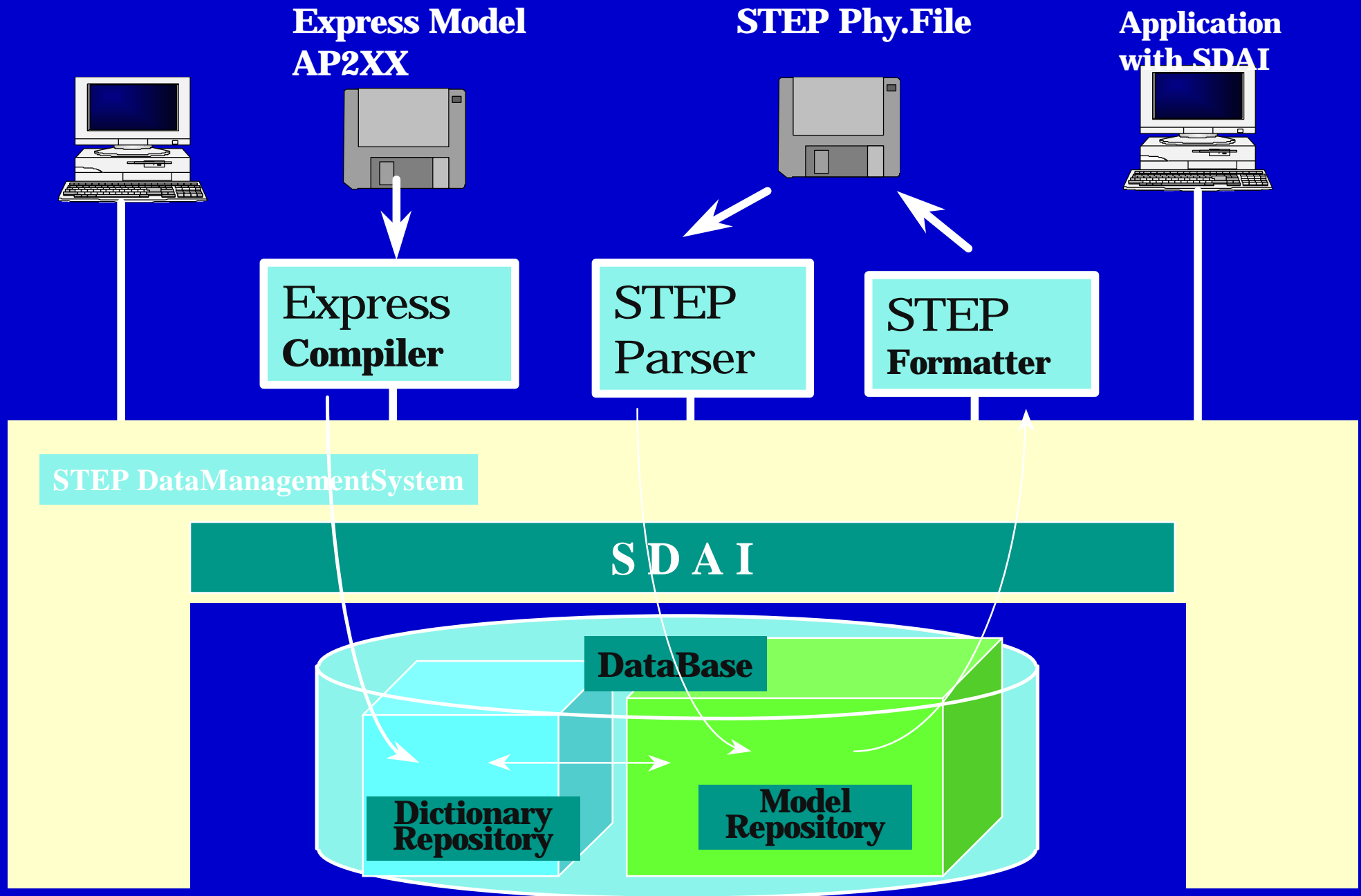


文字インスタンス(Part21ファイル)

```
#420 = TEXT_LITERAL("','TOP VIEW',#430,'baseline left',.RIGHT.,#70);
#430 = AXIS2_PLACEMENT_2D(",#440,$);
#440 = CARTESIAN_POINT(",(1.34976081725825,-1.40961478361176));
#450 = TEXT_STYLE_FOR_DEFINED_FONT(#60);
#460 = TEXT_STYLE_WITH_BOX_CHARACTERISTICS(",#450,(BOX_HEIGHT(3.2),
BOX_WIDTH(3.2),BOX_SLANT_ANGLE(0.),BOX_ROTATE_ANGLE(0.));
#470 = PRESENTATION_STYLE_ASSIGNMENT((#460));
#480 = (
ANNOTATION_OCCURRENCE()
ANNOTATION_TEXT_OCCURRENCE()
DRAUGHTING_ANNOTATION_OCCURRENCE()
REPRESENTATION_ITEM("")
STYLED_ITEM((#470),#420)
GEOMETRIC_REPRESENTATION_ITEM());
#490 = MAPPED_ITEM(",#500,#540);
#500 = REPRESENTATION_MAP(#510,#230);
#510 = AXIS2_PLACEMENT_2D(",#520,#530);
#520 = CARTESIAN_POINT(",(0.,0.));
#530 = DIRECTION(",(1.,0.));
#540 = AXIS2_PLACEMENT_2D(",#550,#560);
#550 = CARTESIAN_POINT(",(10.9074453128387,26.4253893960984));
#560 = DIRECTION(",(1.,0.));
```



STEPの実装とTool Kit



SDAI データベース関連関数

SDAI データベース関連関数:

- sdaiCreateModel()*, *sdaiCreateModelBN()*
- sdaiDeleteModel()*
- sdaiOpenModel()*, *sdaiOpenModelBN()*
- sdaiOpenRepository()*, *sdaiOpenRepositoryBN()*
- sdaiCloseModel()*

SDAI 初期化関数

Databaseにアクセスするには、先ず sessionがオープンされていなければならない

Modelを操作するには、それが属するRepositoryが、先ずオープンされる必要がある

モデルをオープンする場合、しかるべきアクセス権を設定する

関連する関数

–*sdaiOpenSession()*, *sdaiCloseSession()*

–*sdaiOpenRepository()*, *sdaiCloseRepository()*

–*sdaiCreateModel()*

–*sdaiOpenModel()*, *sdaiCloseModel()*, *sdaiDeleteModel()*

SDAI 識別子検索機能

名前で検索する場合とInstance識別子を用いて検索する場合がある

関連する関数

- *sdaiGetEntity()*, *sdaiGetAttrDefinition()*, *sdaiGetAttrDefinitionBN()*

```
SdaiInstance pointId, xId,yId,modelId;
```

```
pointId = sdaiGetEntity(modelId,"point");
```

```
xId = sdaiGetAttrDefinitionBN("MySchema","POINT","X");
```

```
yId = sdaiGetAttrDefinition(pointId,"y");
```

SDAI データ操作関数 I

如何なるデータ操作関数も、名前またはインスタンス識別子を用いて、実際の EXPRES 項目(entityName, attributeName)を識別する

インスタンス識別子を用いた方が性能は良い

データ操作関数は、インスタンスの特定の属性や他のどの属性にもアクセス出来る

関連する *EDMinterface* 関数:

–*sdaiCreateInstance()*, *sdaiDeleteInstance()*

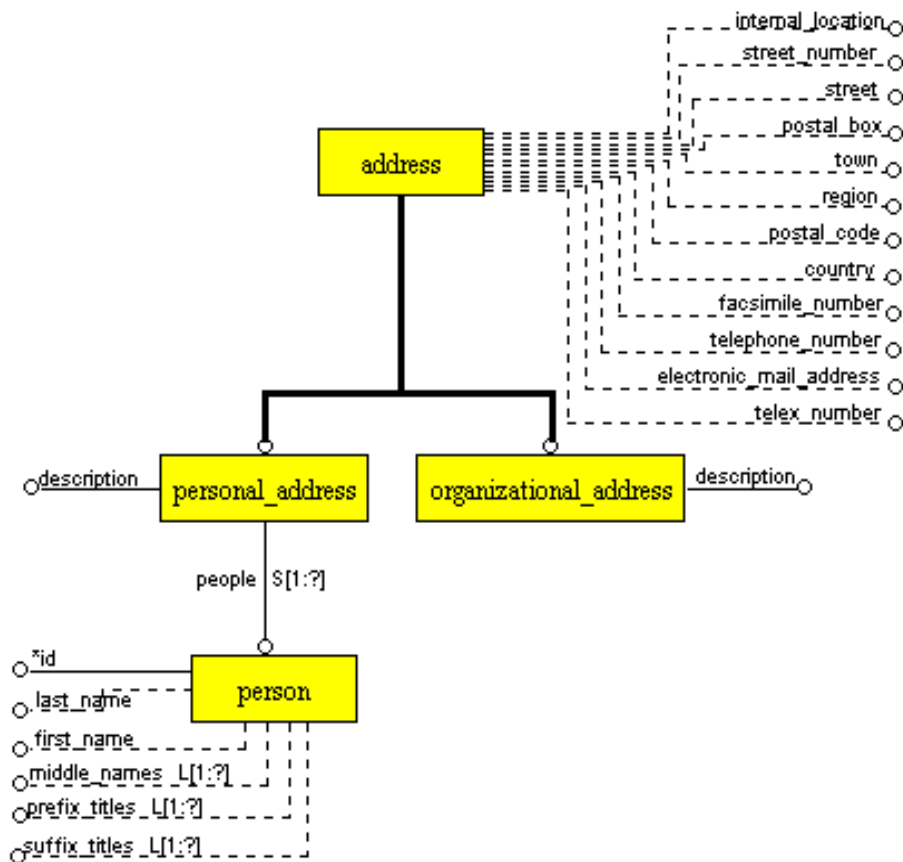
–*sdaiPutAttr()*, *sdaiGetAttr()*, *sdaiPutAttrs()*, *sdaiGetAttrs()*

–*sdaiTestAttr()*, *sdaiUnsetAttr()*

–*sdaiCreateAggr()*, *edmiDeleteAggr()*

–*sdaiCreateNestedAggr()*

SDAIの使用例



```

ENTITY organizational_address
  SUBTYPE OF (address);
  organizations : SET [1:?] OF organization;
  description : text;
END_ENTITY;
  
```

```

ENTITY personal_address
  SUBTYPE OF (address);
  people : SET [1:?] OF person;
  description : text;
END_ENTITY;
  
```

```

ENTITY address;
  internal_location : OPTIONAL label;
  street_number : OPTIONAL label;
  street : OPTIONAL label;
  postal_box : OPTIONAL label;
  town : OPTIONAL label;
  region : OPTIONAL label;
  postal_code : OPTIONAL label;
  country : OPTIONAL label;
  facsimile_number : OPTIONAL label;
  telephone_number : OPTIONAL label;
  electronic_mail_address : OPTIONAL label;
  telex_number : OPTIONAL label;
WHERE
  WR1: EXISTS(internal_location) OR
  EXISTS(street_number) OR
  EXISTS(street) OR
  EXISTS(postal_box) OR
  EXISTS(town) OR
  EXISTS(region) OR
  EXISTS(postal_code) OR
  EXISTS(country) OR
  EXISTS(facsimile_number) OR
  EXISTS(telephone_number) OR
  EXISTS(electronic_mail_address) OR
  EXISTS(telex_number);
END_ENTITY; -- address
  
```

```

ENTITY person;
  id : identifier
  last_name : OPTIONAL label;
  first_name : OPTIONAL label;
  middle_name : OPTIONAL LIST[1:?] OF label;
  prefix_titles : OPTIONAL LIST[1:?] OF label;
  suffix_titles : OPTIONAL LIST[1:?] OF label;
  UNIQUE
  UR1 : id;
WHERE
  WR1 : EXISTS (last_name) OR EXISTS (first_name);
END_ENTITY;
  
```

SDAIの使用例

```
modelRepository = sdaiOpenRepositoryBN ("ModelRepository");
model = sdaiOpenModelBN (modelRepository, "EmployeeAddress", sdaiRO);
aggeregate = sdaiGetEntityExtentBN (model, "personal_address");
MaxNumber = sdaiGetMemberCount (aggregate);
Iterator = sdaiCreateIterator (aggregate);
for (i1= 0; i1<MaxNumber; i1++) {
    sdaiNext (Iterator);
    sdaiGetAggrByIterator (Iterator, sdai INSTANCE, &resultId);
    sdaiGetAttrBN (resultId, "internal_location", sdaiSTRING, &intLocation);
    sdaiGetAttrBN (resultId, "street_number", sdaiSTRING, &streetNo);
    sdaiGetAttrBN (resultId, "street", sdaiSTRING, &street);
    sdaiGetAttrBN (resultId, "postal_box", sdaiSTRING, &boxNo);
    sdaiGetAttrBN (resultId, "town", sdaiSTRING, &town);
    sdaiGetAttrBN (resultId, "region", sdaiSTRING, &region);
    sdaiGetAttrBN (resultId, "postal_code", sdaiSTRING, &postalCode);
    sdaiGetAttrBN (resultId, "country", sdaiSTRING, &country);
    sdaiGetAttrBN (resultId, "facsimile_number", sdaiSTRING, &faxNumber);
    sdaiGetAttrBN (resultId, "telephone_number", sdaiSTRING, &phoneNumber);
    sdaiGetAttrBN (resultId, "electronic_mail_address", sdaiSTRING, &e-mailAddress);
    sdaiGetAttrBN (resultId, "telex_number", sdaiSTRING, &telexNumber);
    sdaiGetAttrBN (resultId, "description", sdaiSTRING, &description);
    sdaiGetAttrBN (resultID, "people", sdaiAGGR, & AggrID);
    nElements = sdaiGetMemberCourt (AggrId);
    IterId = sdaiCreateIterator (AggrId);
    for (i2 = 0; i2<nElements; i2++) {
        sdaiNext (IterId);
        sdaiGetAggrByIterator (IterId, sdai INSTANCE, &InstanceId);
        sdaiGetAttrBN (InstanceId, "id", sdai STRING, &id);
        sdaiGetAttrBN (InstanceId, "last_name", sdai STRING, &lastName);
        sdaiGetAttrBN (InstanceId, "first_name", sdaiSTRING, &firstName);
        .
        .
        .
    }
}
```

:モデル名EmployeeAddressをオープンする
:エンティティpersonal_addressにアクセス
:personal_addressエンティティのインスタンスの数

:インスタンスを1ヶ毎呼び出して属性値を得る

: Internal location (地域名)

: Street number (通り番号)

: Street (通り)

: Postal box (私書箱)

: Town (町)

: region (地域)

: Postal code (郵便番号)

: Country (国)

: Fax (ファクス番号)

: Phone (電話番号)

: E-mail (eメールアドレス)

: Telex (テレックス番号)

: 属性peopleが指すpersonエンティティを指定

: personエンティティのインスタンスの数

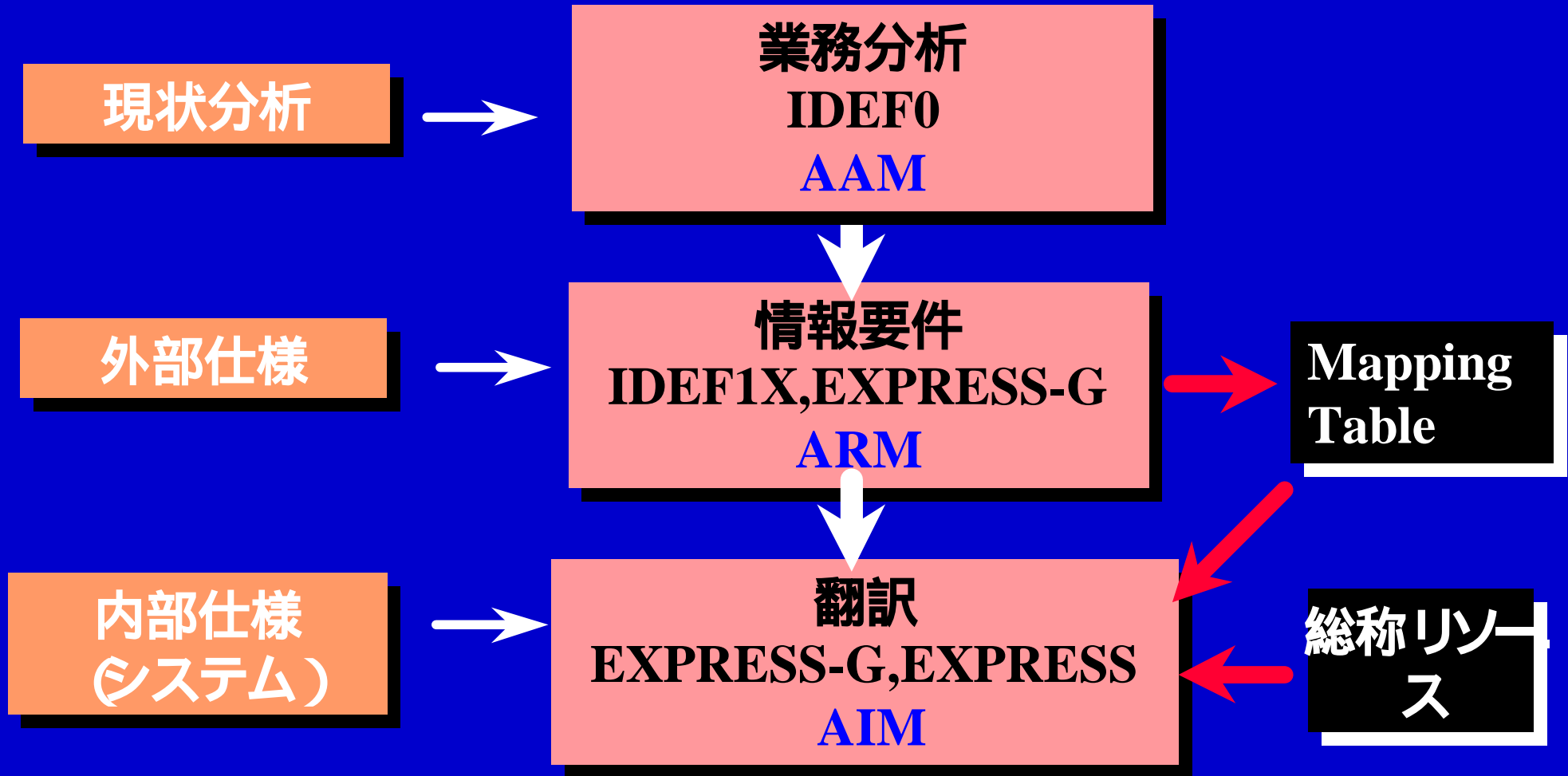
: 同一住所に住む人の名前を取り出す

: 識別番号

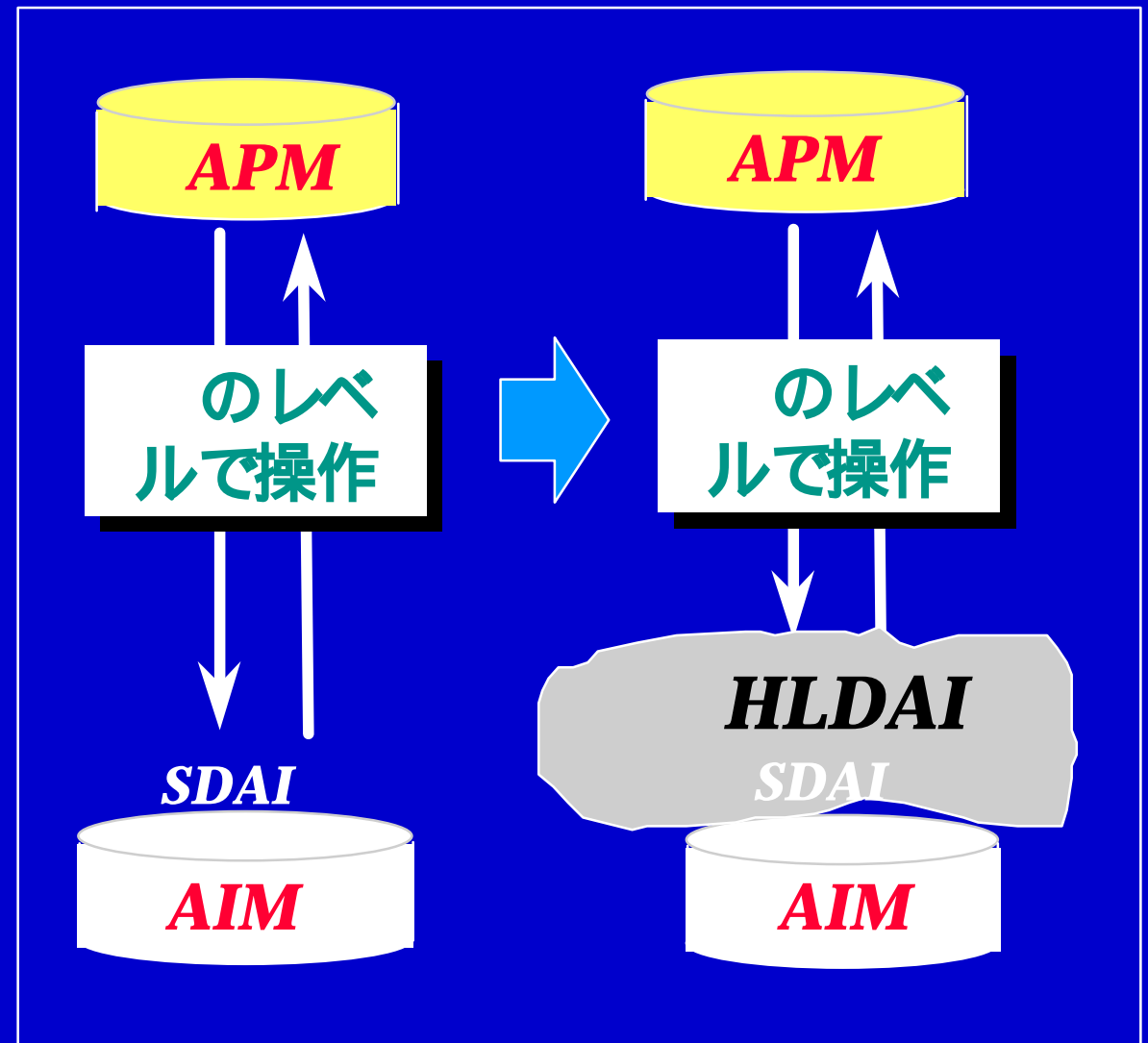
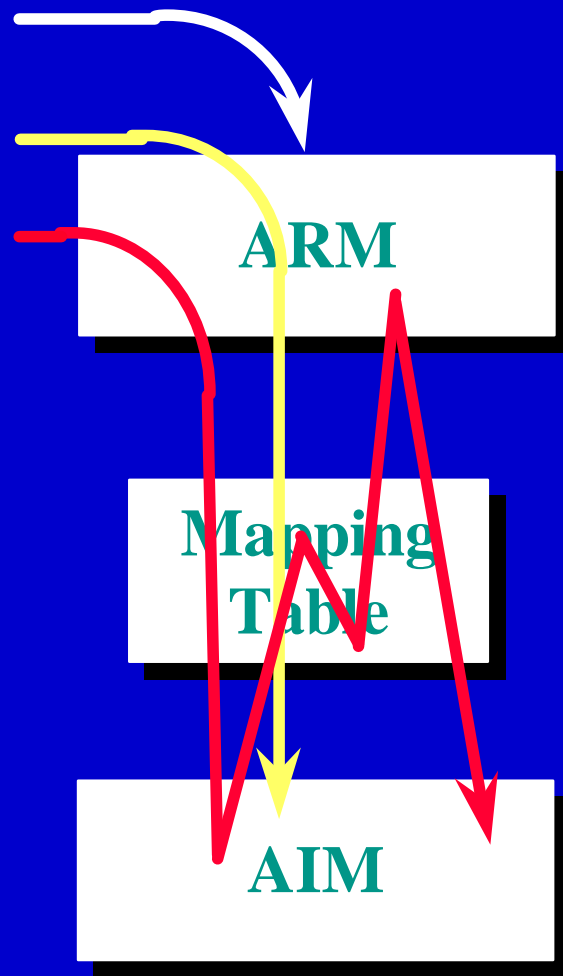
: 姓

: 名

STEPの仕組み



ARMとAIMのギャップを埋めるHLDAI



HLDAIとMapping

プロジェクト概要

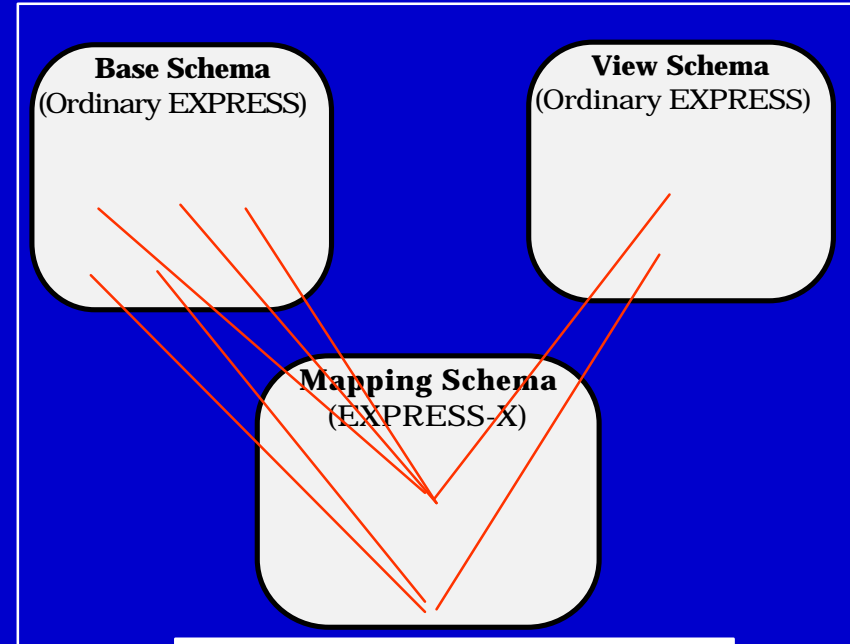
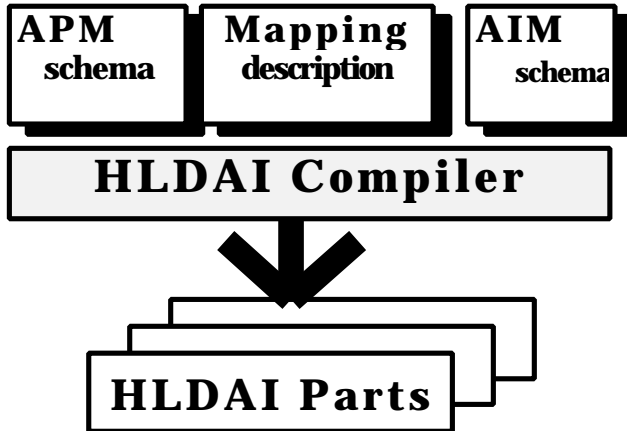
内容

- ・STEPデータへのアクセスを業務レベルで可能にする仕組みを提供する。

狙い

- ・「より効率的にSTEPシステムの構築が行える環境」を整備する

HLDAI Generator



Mapping Language

EXPRESS-Xは、ISOのNWI

・Dr.Hardwick (RPI), Ian Bailey (EuroSTEP)

EXPRESS-Xは、異なるスキーマ間のデータ交換プログラムのコンパイラの仕様

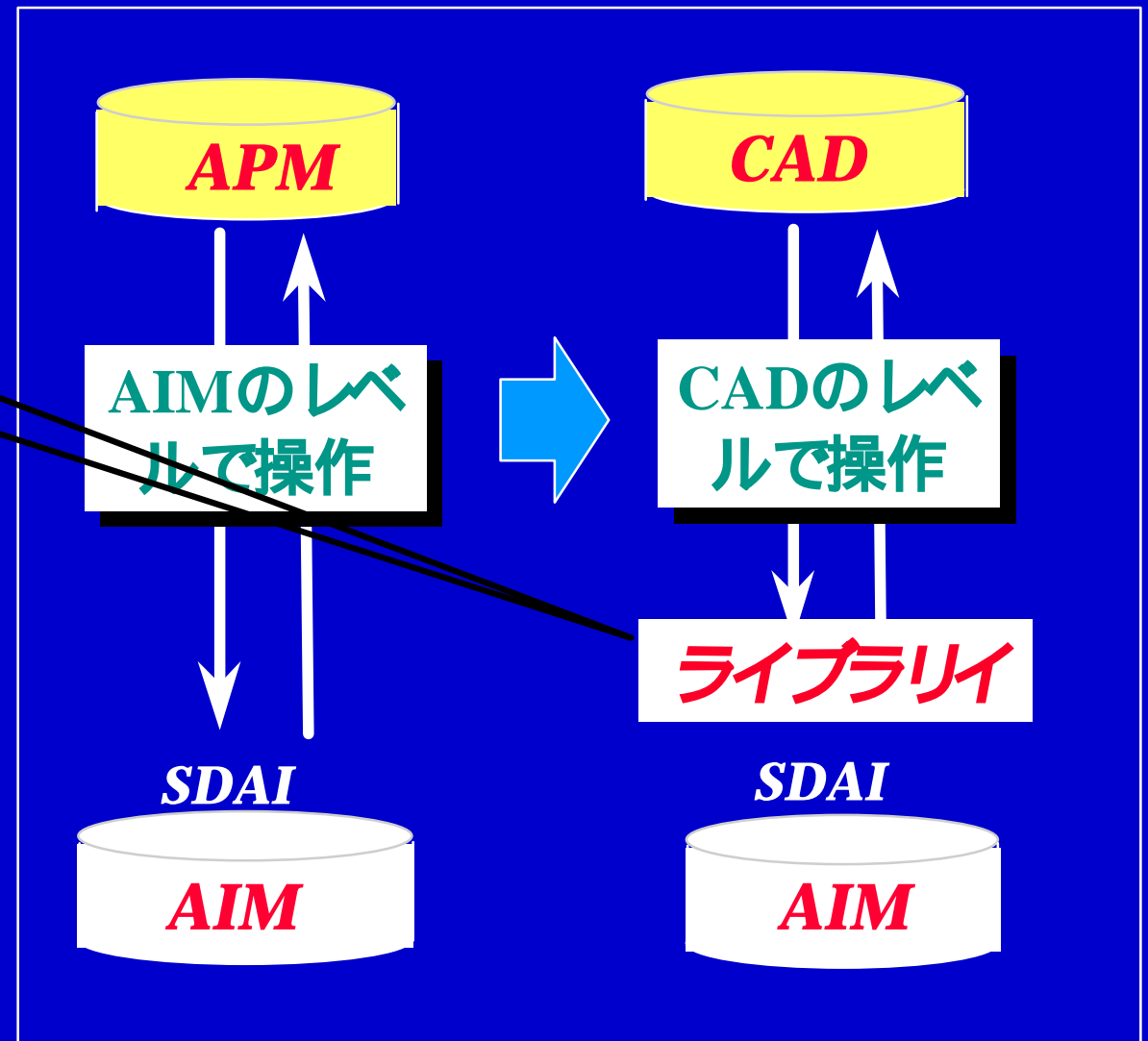
一括変換のイメージが強く、過渡的な処理には不向き。

EXPRESS-Xの源は、EXPRESS-V(TOOLs Inc)とEXPRESS-M(CIMIO)

建設CALISの共通ライブラリ

CADに限定した効率的なアクセスルーチン群を準備

STEPをライブラリで隠蔽して利用者はCADのビューでハンドリング出来る



主な共通ライブラリ

下記のそれぞれにCADのビューで関数群を準備する

1.幾何要素

- 点、直線、折れ線、円・円弧、楕円・楕円弧

2.表記要素

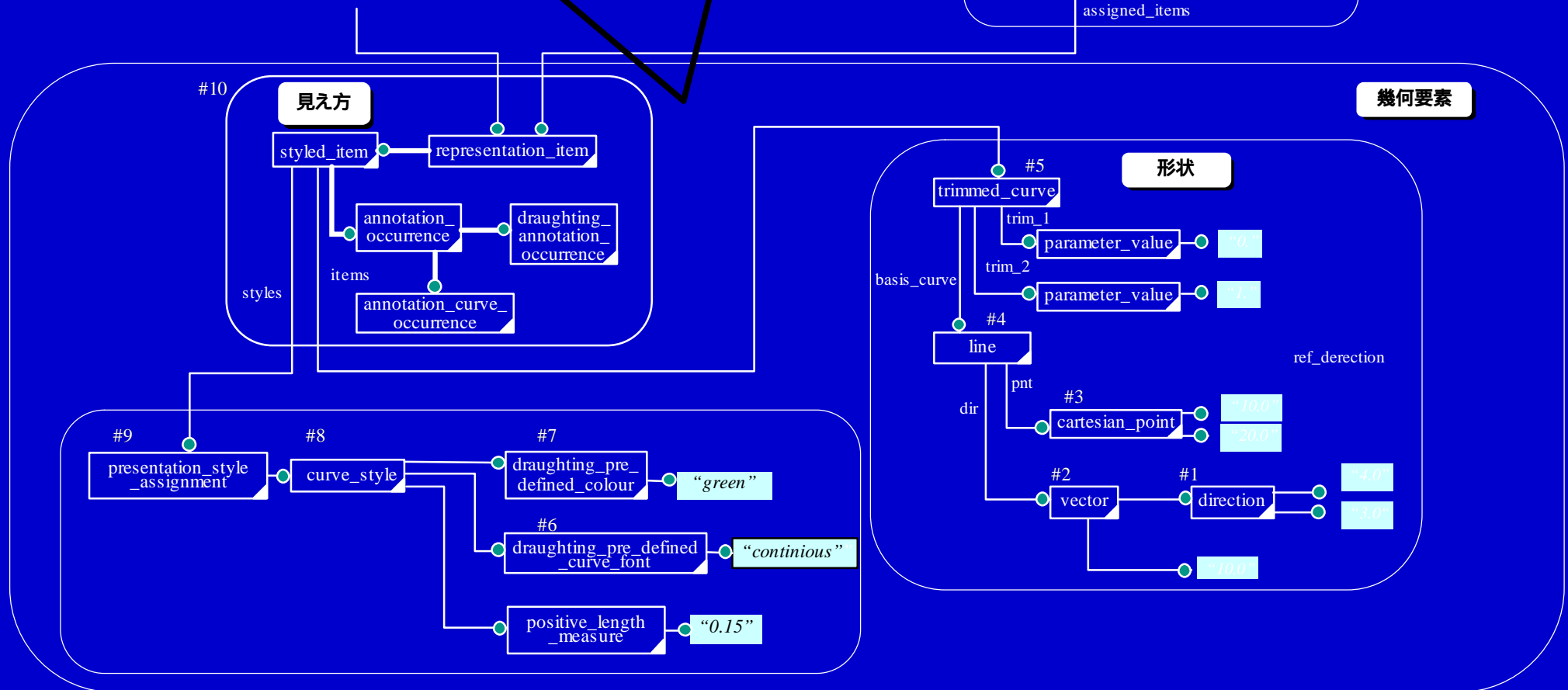
- 文字
- 寸法線（直線寸法、角度寸法、半径・直径寸法等）
- ハッチング等

3.共通

- STEPファイル読み書き

主な共通ライブラリ --- 直線の場合のイメージ

```
cadLine(layer, view, color, lineFont, line  
Width, x1, y1, x2, y2, &return);
```



共通ライブラリの使用イメージ

(1) Read時 :

```
return = OpenPart21 ("FileName", "RO", level);
```

```
if (return != 0){... ..}
```

```
GetElementTypes (ElementTypeArray);
```

```
while(ElementTypeArray[i1]) {
```

```
    ReadElementIds (ElementTypeArray[i1],ElementIds); 構成要素の識別子
```

```
    while (ElementIds[i2]) {
```

```
        GetElementMember (ElementIds[i2],structure);
```

```
        case ElementIds[i2]:
```

```
            .
```

```
            .
```

(2) Write時 :

```
return = OpenPart21 ("FileName", "RW", level);
```

```
if (return != 0) { ... .. }
```

```
PutElementMember(ElementType,structure);
```

```
    .
```

```
    .
```

:Part21 ファイルをオープンする

:Error処理をする

:ElementType(Feature)全体を得る

:ElementType毎に処理する

:構成要素の識別子

:構成要素毎に処理をする

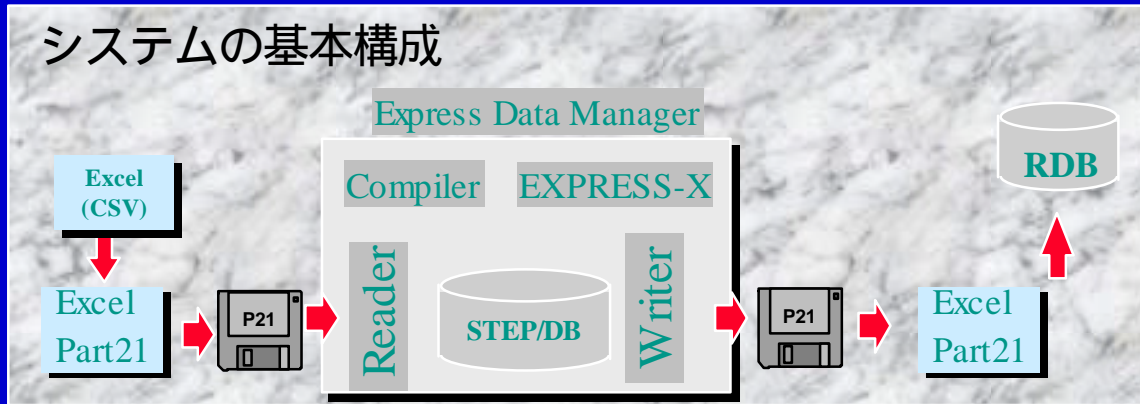
: Part21 ファイルをオープンする

: Error処理をする

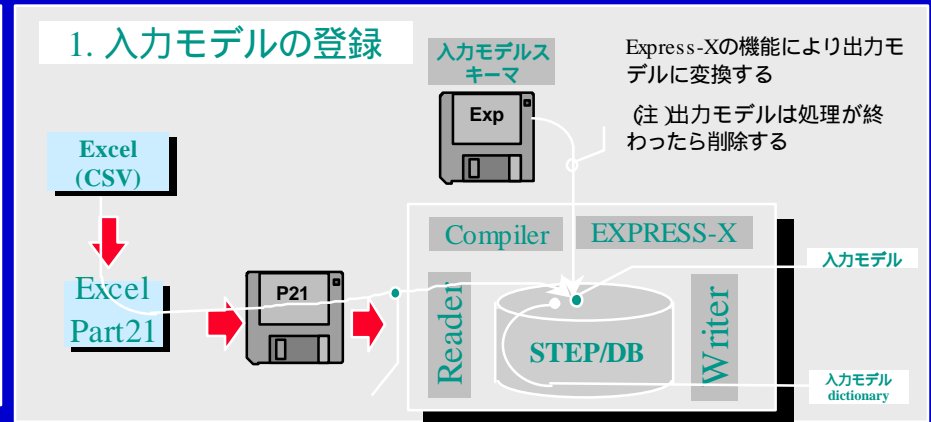
: ElementType(Feature)毎に処理

EXPRESS-XによるSTEPデータの扱い

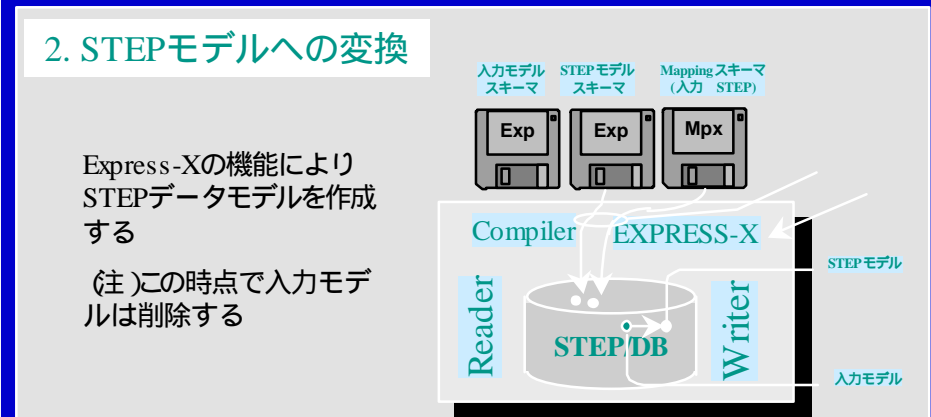
システムの基本構成



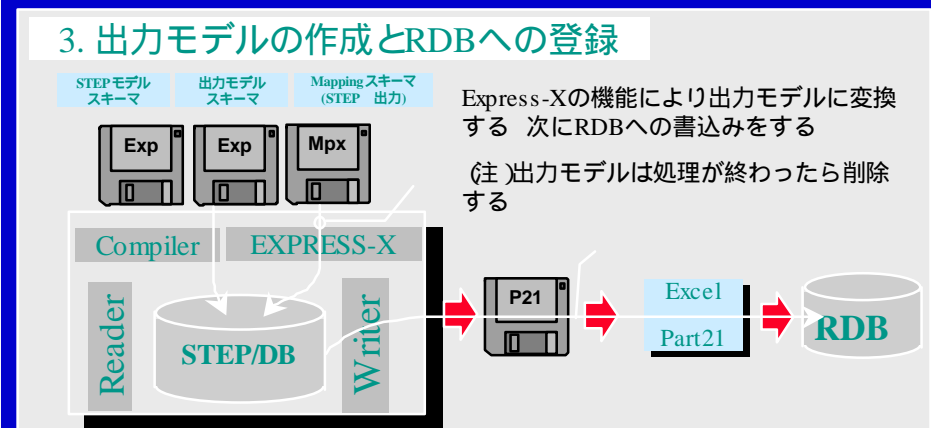
1. 入力モデルの登録



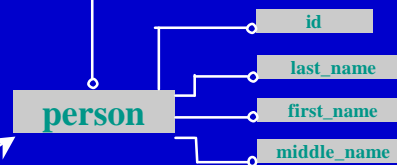
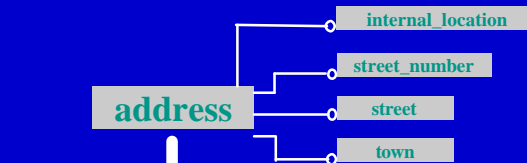
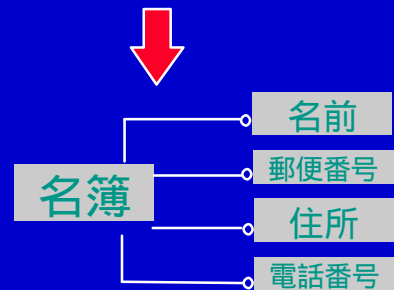
2. STEPモデルへの変換



3. 出力モデルの作成とRDBへの登録



名前	郵便	住所	電話
鈴木 一郎	00	東京都...	03-5555-xxxx
山田 太郎	12	福岡市...	092-812-yyyy



ISO10303Part40番台のリリースを用いて表現

入力モデル

CSV表形式モデルからSTEPモデルに変換するために1次的に作成する (STEP形式の)モデル

STEPモデル

維持される(貯蔵用の)STEPデータモデル 国際標準に基づくデータ表現がされている

計算機が解釈可能な製品データの表現方法 (構造と制約) と交換方法の規定
製品のライフサイクルを通じた製品DBの実装、共有、保管の基盤を提供する

ISO10303 Industrial automation system and integration - Product data representation and exchange

物を表現するSTEPの偉大な力とその活用に取り組みよう!

参考文献など

- 木村他：製品モデルの表現とその利用法 』 日本規格協会
国内で始めて出版されたSTEPの本、AP201の実装を元に解説。(¥2,900)
- J.Fowler: 『STEPがわかる本 』 工業調査会
プラントCALS研究会が翻訳。STEPの全貌を記述している。(¥2,800)
- 富士通：『STEP/EXPRESSへの道 』 富士通STEP研究開発センター
STEPシステムとペアになった自己学習教材。(¥19,800)
- D. Schenck他：『Information Modeling the EXPRESS way 』
Oxford
EXPRESSによる製品のモデリングを詳細に解説している。(US\$ 50.00)