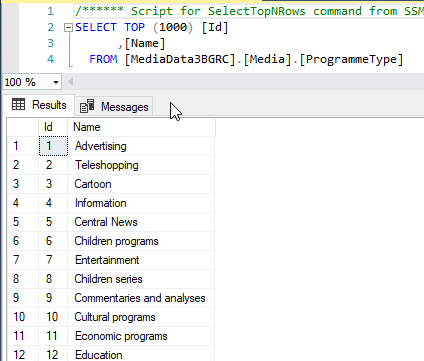
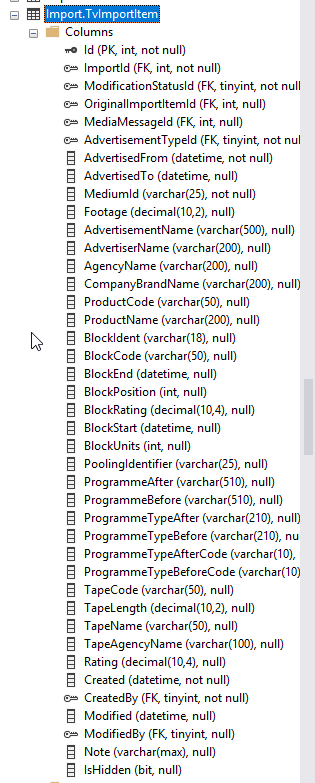
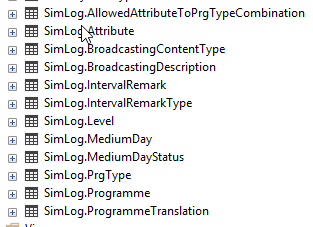
Kompozitní pořad před/v/po



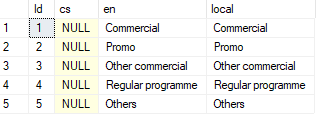
Tabulky SimLogu



**AllowedAttribute** many to many tabulka



**Attribute** ciselnik



**BroadcastingContentType**  ciselnik



**BroadcastingDescription** vyskytyporadu



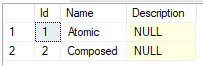
**IntervalRemark** znacky pro Zachytavadlo, aby holky vedely kde jsou reklamy



**IntervalRemarkType** ciselnik, jen jedna hodnota



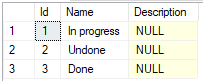
**Level** ciselnik atomic composite



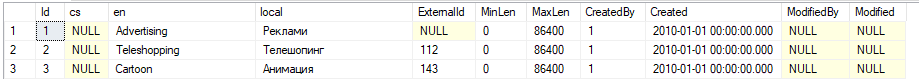
**MediumDay** jeden vysilaci den den jednoho media. Zacina o pulnoci, při nacitani v SimLogu tototo dne ale dostanu streamy od 6. hodiny do 6. hodiny nasledujiciho dne.



**MediumDayStatus** ciselnik



**PrgType** ciselnik, typ programu



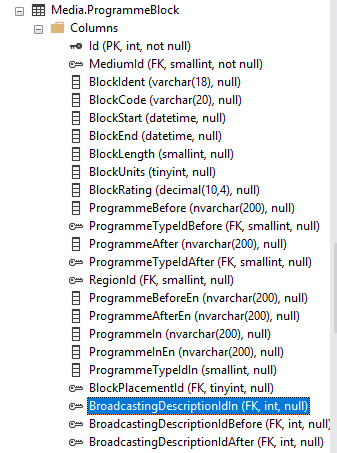
**Programme –** program napr televizní noviny. Nejedna se o vyskyt ale o definici , ma nejaky typ – prgTypeId



**ProgrammeTranslation**

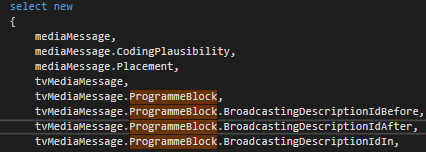


**Media.ProgrammeBlock - pribyly tri dolni sloupce**



**Pridani ProgrammeDatacontractu do query:**

**TvMediaMessageDao.**



var data = queryable.ToArray();

List<int> programs = data.Where(d => d.BroadcastingDescriptionIdBefore != null).Select(d => d.BroadcastingDescriptionIdBefore.Value).ToList();

programs.AddRange(data.Where(d => d.BroadcastingDescriptionIdIn != null).Select(d => d.BroadcastingDescriptionIdIn.Value));

programs.AddRange(data.Where(d => d.BroadcastingDescriptionIdAfter != null).Select(d => d.BroadcastingDescriptionIdAfter.Value));

List<ProgrammeDataContract> programmeDataContracts = (from bd in model.BroadcastingDescription

join pr in model.Programme on bd.ProgrammeId equals pr.Id

join prgType in model.PrgType on pr.PrgTypeId equals prgType.Id

join tra in model.ProgrammeTranslation on pr.Id equals tra.TranslatedEntityId

where programs.Contains(bd.Id)

select new ProgrammeDataContract

{

BroadcastingDescriptionId = bd.Id,

ProgrammeName = pr.Name,

ProgrammeNameEn = tra.Name,

ProgrammeTypeId = prgType.Id,

ProgrammeType = prgType.Local,

ProgrammeTypeEn = prgType.En

}).ToList();

foreach (var a in data)

{

}

if (a.ProgrammeBlock != null)

{

message.ProgrammeBlock = DirectMap.ObjectToObject<ProgrammeBlock>(a.ProgrammeBlock);

message.ProgrammeBlock.ProgrammeBefore = programmeDataContracts.FirstOrDefault(d => d.BroadcastingDescriptionId == message.ProgrammeBlock.BroadcastingDescriptionIdBefore);

message.ProgrammeBlock.ProgrammeIn = programmeDataContracts.FirstOrDefault(d => d.BroadcastingDescriptionId == message.ProgrammeBlock.BroadcastingDescriptionIdIn);

message.ProgrammeBlock.ProgrammeAfter = programmeDataContracts.FirstOrDefault(d => d.BroadcastingDescriptionId == message.ProgrammeBlock.BroadcastingDescriptionIdAfter);

message.ProgrammeBlock.AcceptChanges();

}