业务类整理

1.

查找王贵芳的记录，调用ICC的时候，key值是

http://192.168.10.77:9200/iCCWebClient/Common/Connect?key=bf6c6290-e265-4fab-9bf6-cc238fd7aece

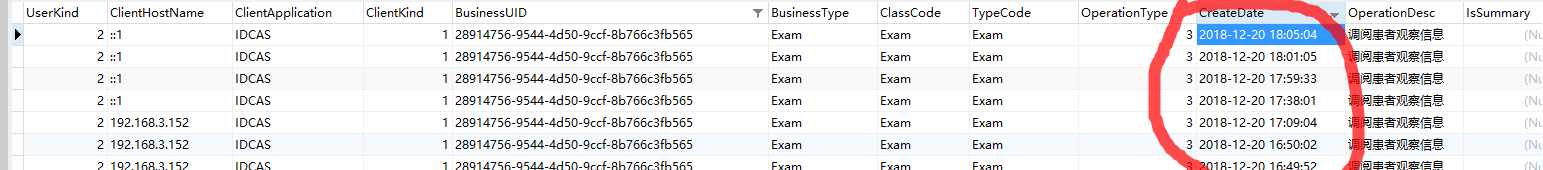


根据姓名去找到这个key bf6c6290-e265-4fab-9bf6-cc238fd7aece

发现每一次调阅查看改归档信息，就会往数据库里增加一条记录。

\_logClinicDataOperationRepository.Insert(log);

反应到数据库

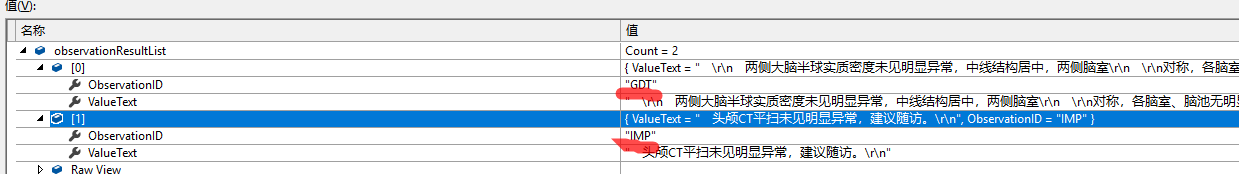


发现查看影像的时候提到了苗哥的ICC当中。

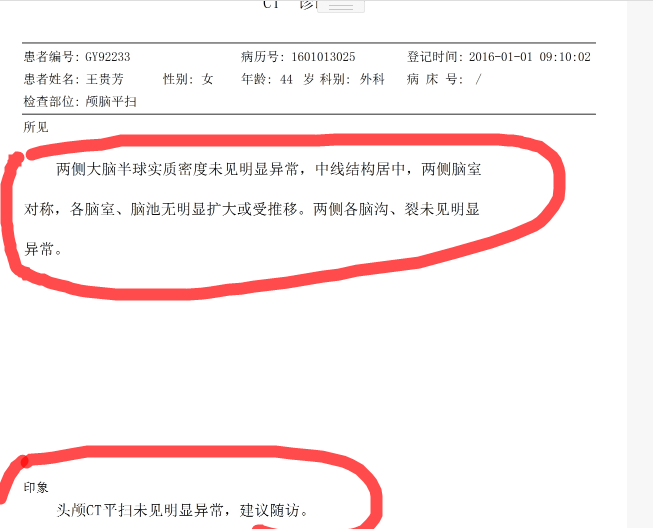
去远程服务器上发现，云归档和苗哥那边的IIC是部署在一起的。

其中ObservationRequest（观察请求记录）和ObservationResult（观察结果表）是一对多，就像下面，同一个观察，会有两个结果。

一个是所见，一个是印象。

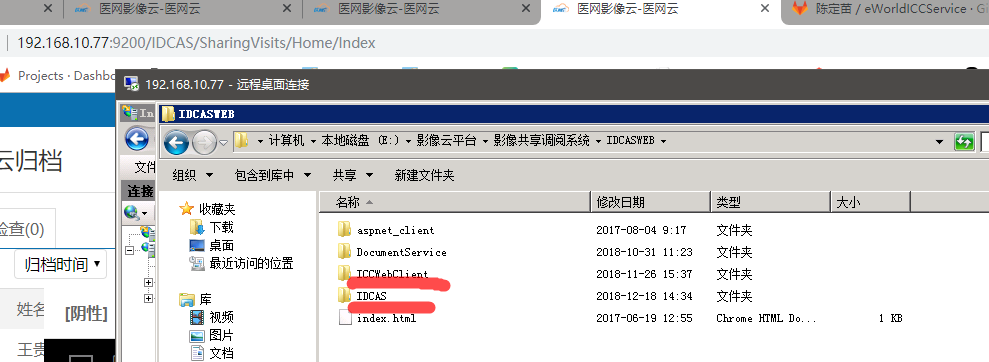


反馈到报告单子上：



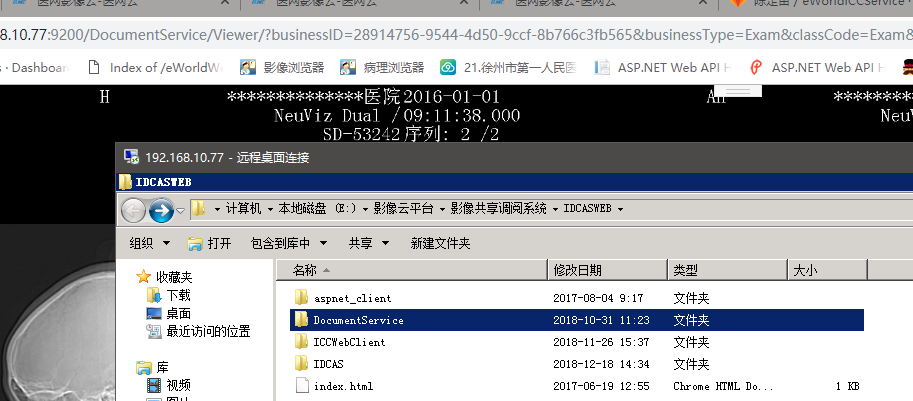
观察数据库发现成对出现，一格式影像所见，一个是影像诊断。



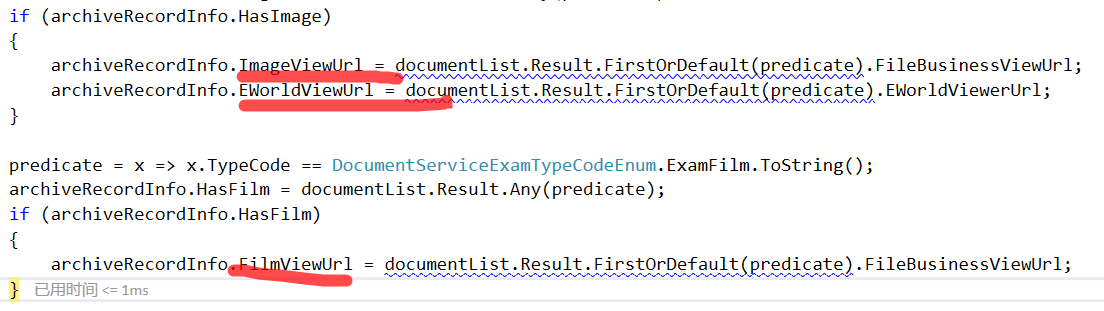


<http://192.168.10.77:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&businessType=Exam&classCode=Exam&typeCode=ExamFilm&fileUid=6d58c036-271d-477b-9d20-35a014f851fe&organizationID=2470&print=&download=&businessView=true&token=3608a9972db823efa9fb268e3d42a110&reportimgpost=&Anonymous=>

这类参数，连接到当前文件夹的另一个项目。



各种地址的获取：



入参：

{

"ObservationUID":"28914756-9544-4d50-9ccf-8b766c3fb565",

"PatientID":"GY92233"

}

结果就是：

{

"state": true,

"msg": "OK",

"data": {

"ObservationUID": "28914756-9544-4d50-9ccf-8b766c3fb565",

"Name": "王贵芳",

"Sex": "女",

"Age": 44,

"AgeUnit": "岁",

"ClinicInfoType": "Exam",

"PatientMasterID": "8ab73c3f-0aa8-4e9c-95fa-3ae79d4522cd",

"PointOfCare": "",

"PatientClass": "普通",

"InPatientNO": "",

"Bed": "",

"MedRecNO": "SD-1601013025",

"RelevantClinicalInfo": "",

"ClinicDiagnosis": "",

"Attention": "",

"ProviderName": "陈业祥",

"RequestDeptName": "外科",

"RequestedDate": "/Date(1513126978000)/",

"HasResult": true,

"RepetorViewUrl": "http://localhost:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&businessType=Exam&classCode=Exam&typeCode=ExamResult&fileUid=46361560-7e57-456c-87ce-33a71afa5b73&organizationID=2470&print=&download=&businessView=true&token=2bd17495d9c2f6894188458acbcbec7a&reportimgpost=&Anonymous=",

"HasRequest": false,

"RequestViewUrl": null,

"HasFilm": true,

"FilmCount": 5,

"FilmViewUrl": "http://localhost:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&businessType=Exam&classCode=Exam&typeCode=ExamFilm&fileUid=6d58c036-271d-477b-9d20-35a014f851fe&organizationID=2470&print=&download=&businessView=true&token=3608a9972db823efa9fb268e3d42a110&reportimgpost=&Anonymous=",

"HasImage": true,

"ImageViewUrl": "http://localhost:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&businessType=Exam&classCode=Exam&typeCode=ExamImage&fileUid=a03e83b4-e8d8-40be-a151-fc44feb163b3&organizationID=2470&print=&download=&businessView=true&token=0c9936e111e6617a8623e45877186e60&reportimgpost=&Anonymous=",

"EWorldViewUrl": "eworldview:FileSrc=Network,IP=192.168.2.35,Port=1004,AE=PACS,StudyInstanceUID=1.2.156.14702.1.1.0.20160101091002000,PatientID=SD-53242,AccessionNumber=GY92233,Modality=CT",

"Reason": "",

"AccessionNumber": "SD-53242",

"ServiceSectID": "CT",

"ObservationDate": "/Date(1545045002000)/",

"ProcedureName": "颅脑平扫",

"ServiceText": "常规CT",

"OutPatientNO": "",

"DataSource": "RIS",

"DataSourceName": "放射科",

"ResultAssistantName": "郭明华",

"ResultDate": "/Date(1545098536000)/",

"ResultReviseName": "",

"AuditDate": "/Date(1545038554000)/",

"StudyDescription": " \r\n 两侧大脑半球实质密度未见明显异常，中线结构居中，两侧脑室\r\n \r\n对称，各脑室、脑池无明显扩大或受推移。两侧各脑沟、裂未见明显\r\n \r\n异常。\r\n",

"DiagnosticImpression": " 头颅CT平扫未见明显异常，建议随访。\r\n",

"Room": "",

"ResultStatus": "审核完成",

"AbnormalFlags": "阴性"

}

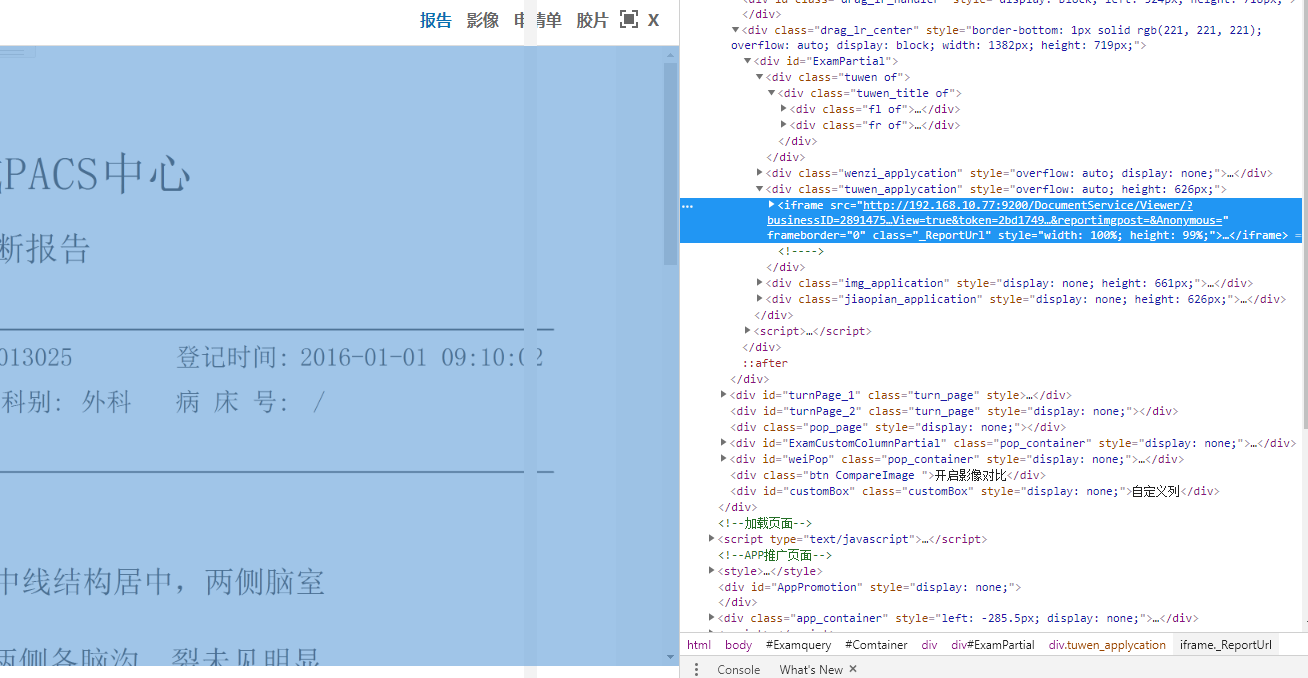
}

网页的结果是：



继续查看网页源代码：

报告这里是一个Iframe标签

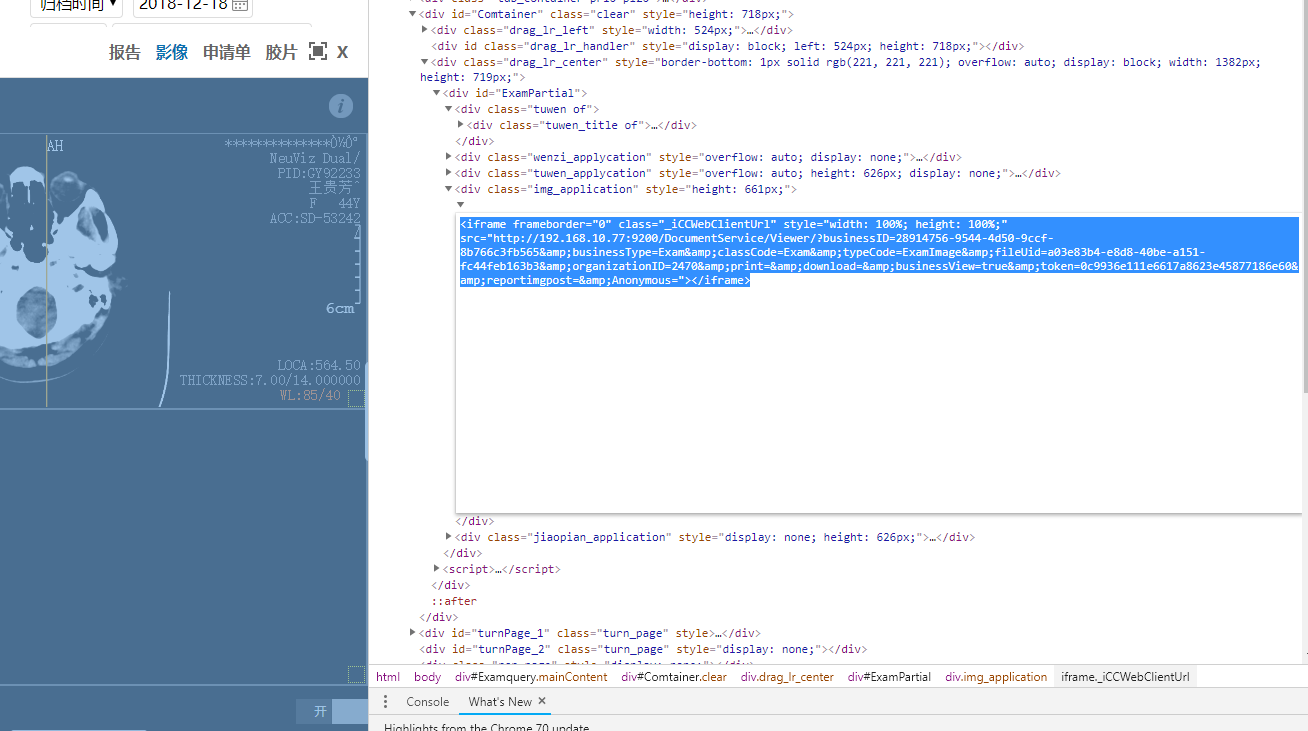


<iframe src="http://192.168.10.77:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&amp;businessType=Exam&amp;classCode=Exam&amp;typeCode=ExamResult&amp;fileUid=46361560-7e57-456c-87ce-33a71afa5b73&amp;organizationID=2470&amp;print=&amp;download=&amp;businessView=true&amp;token=2bd17495d9c2f6894188458acbcbec7a&amp;reportimgpost=&amp;Anonymous=" frameborder="0" class="\_ReportUrl" style="width: 100%; height: 99%;"></iframe>

http://192.168.10.77:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&businessType=Exam&classCode=Exam&typeCode=ExamResult&fileUid=46361560-7e57-456c-87ce-33a71afa5b73&organizationID=2470&print=&download=&businessView=true&token=2bd17495d9c2f6894188458acbcbec7a&reportimgpost=&Anonymous=

这里的businessID就是当前的businessID

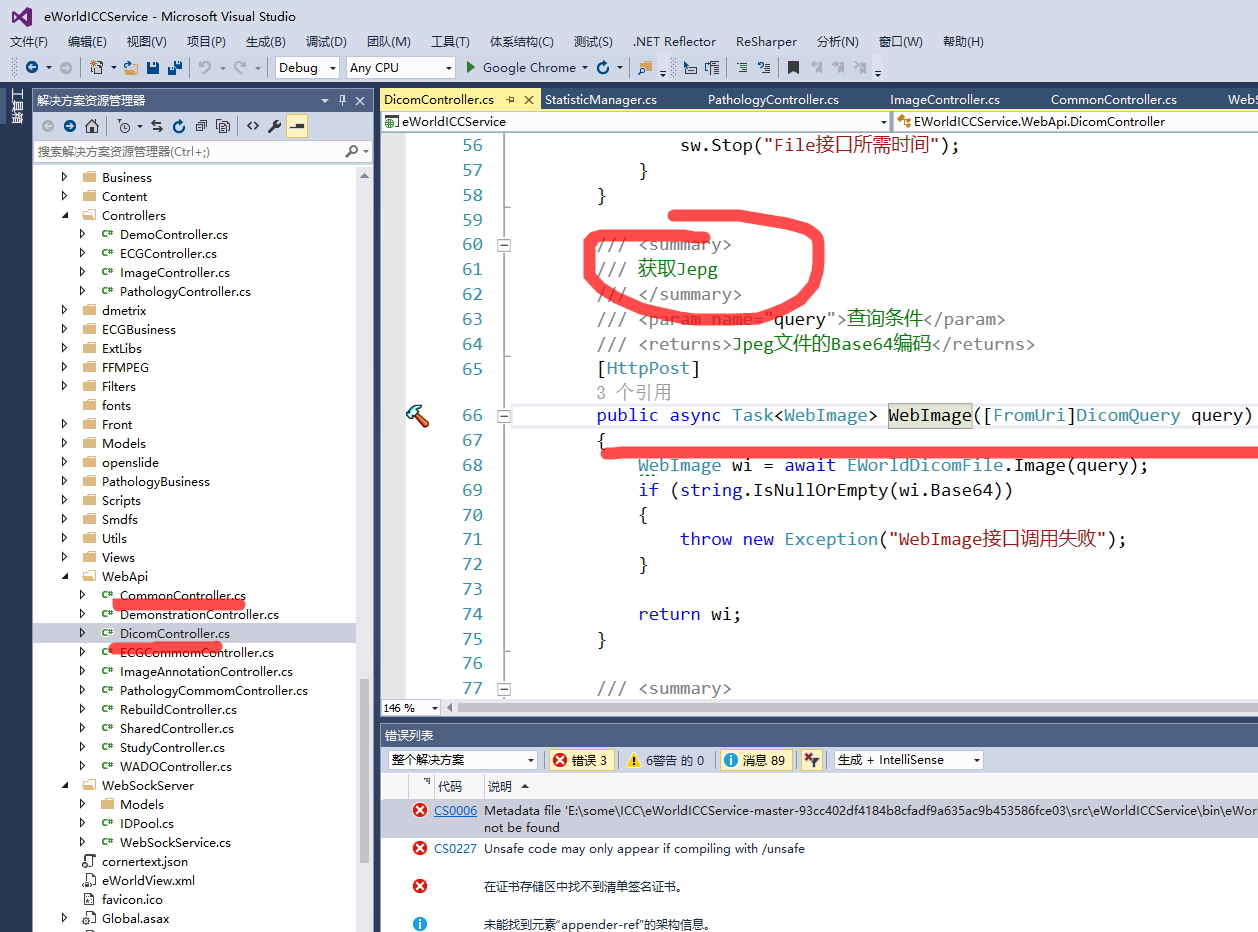
接下来是影像：



也是Iframe标签。

<iframe frameborder="0" class="\_iCCWebClientUrl" style="width: 100%; height: 100%;" src="http://192.168.10.77:9200/DocumentService/Viewer/?businessID=28914756-9544-4d50-9ccf-8b766c3fb565&amp;businessType=Exam&amp;classCode=Exam&amp;typeCode=ExamImage&amp;fileUid=a03e83b4-e8d8-40be-a151-fc44feb163b3&amp;organizationID=2470&amp;print=&amp;download=&amp;businessView=true&amp;token=0c9936e111e6617a8623e45877186e60&amp;reportimgpost=&amp;Anonymous="></iframe>

接下来就走入了苗哥的世界，icc。调了其中的api方法。



2.

注册检查信息：

ObservationRequest（观察请求记录）的OrganizationID（机构号）和crm的机构表（institution）中的InstitutionlCode（机构代码）匹配上：





3.

{

var internetEndpoint = service.UseTLS == false ? service.EndpointInternet : service.EndpointTLSInternet;

if (!string.IsNullOrWhiteSpace(internetEndpoint))

{

\_log.Debug("检测到配置表中有相应外网地址，外网地址=" + internetEndpoint);

result.InternetEndpoint = internetEndpoint;

}

var intranetEndpoint = service.UseTLS == false ? service.Endpoint : service.EndpointTLS;

if (!string.IsNullOrEmpty(intranetEndpoint))

{

\_log.Debug("检测到配置表中有相应内网地址，内网地址=" + intranetEndpoint);

result.IntranetEndpoint = intranetEndpoint;

}

if (string.IsNullOrWhiteSpace(internetEndpoint) && string.IsNullOrWhiteSpace(intranetEndpoint))

{

\_log.Debug("未检测到相应外网地址和内网地址，使用web服务地址作为内网地址。");

result.IntranetEndpoint = GetDocumentServiceUrl();

return result;

}

这里明明是内外网地址找到了

else

{

\_log.Error("外网地址，内网地址，网站地址都没找到或不可用。");

return result;

}

}

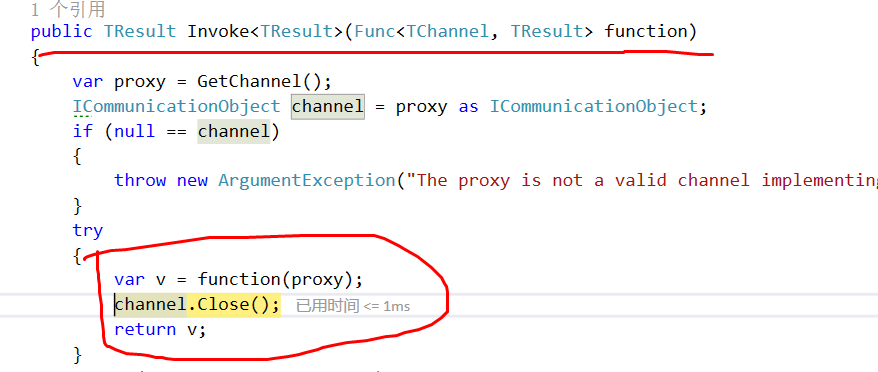
4.

听曹兵电话，湖州是特殊，云是别人的，只能给他身份证号这种，然后去他们的接口请求，返回了影像浏览的地址。（说是长兴，还有哪，台州来着，都是单独部署的）

我得确定，我这边能力平台，你给我的全是localhost开头的地址，我怎么找到影像的具体地址。

5.

wcf的调用，调用完关闭频道。



6.

计时器监视的方法

var swcout = new Stopwatch();

swcout.Start();

\_log.Info("app获取调阅检查详情执行数据库查询开始");

var resultdata = \_imageArchiveClient.Invoke(t => t.GetPatientExamInfo(query.ToQuery()));

swcout.Stop();

\_log.Info($"app获取调阅检查详情执行数据库查询结束，耗时{swcout.ElapsedMilliseconds}毫秒");

if (resultdata.code == ResultCodeEnum.Success)

{

//swcout.Restart();

//\_log.Info("app获取调阅检查详情执行ToDTO方法开始");

data = resultdata.Result.ToDTO();

//swcout.Stop();

//\_log.Info($"app获取调阅检查详情执行oDTO方法结束，耗时{swcout.ElapsedMilliseconds}毫秒");

//data.Name = DESHelper.Encrypt(data.Name);

//data.Sex = DESHelper.Encrypt(data.Sex);

//data.OrganizationName = DESHelper.Encrypt(data.OrganizationName);

//data.OrganizationID = DESHelper.Encrypt(data.OrganizationID);

//data.AccessionNumber = DESHelper.Encrypt(data.AccessionNumber);

//swcout.Restart();

//\_log.Info("app获取调阅检查详情执行数据转换1开始");

//var url = ConfigDefine.Url;

//if (data.HasResult.HasValue && data.HasResult.Value)

//{

// data.RepetorViewUrl = CommonUtil.TransLocalhostForClient(data.RepetorViewUrl, url);

//}

//swcout.Stop();

//\_log.Info($"app获取调阅检查详情执行数据1转换，耗时{swcout.ElapsedMilliseconds}毫秒");

//swcout.Restart();

//\_log.Info("app获取调阅检查详情执行数据转换2开始");

//swcout.Stop();

//\_log.Info($"app获取调阅检查详情执行数据2转换，耗时{swcout.ElapsedMilliseconds}毫秒");

//swcout.Restart();

//\_log.Info("app获取调阅检查详情执行数据转换3开始");

//if (data.HasImage.HasValue && data.HasImage.Value)

//{

// data.ImageViewUrl = CommonUtil.TransLocalhostForClient(data.ImageViewUrl, url);

//}

//swcout.Stop();

//\_log.Info($"app获取调阅检查详情执行数据3转换，耗时{swcout.ElapsedMilliseconds}毫秒");

//swcout.Restart();

//\_log.Info("app获取调阅检查详情执行数据转换4开始");

//if (data.HasFilm.HasValue && data.HasFilm.Value)

//{

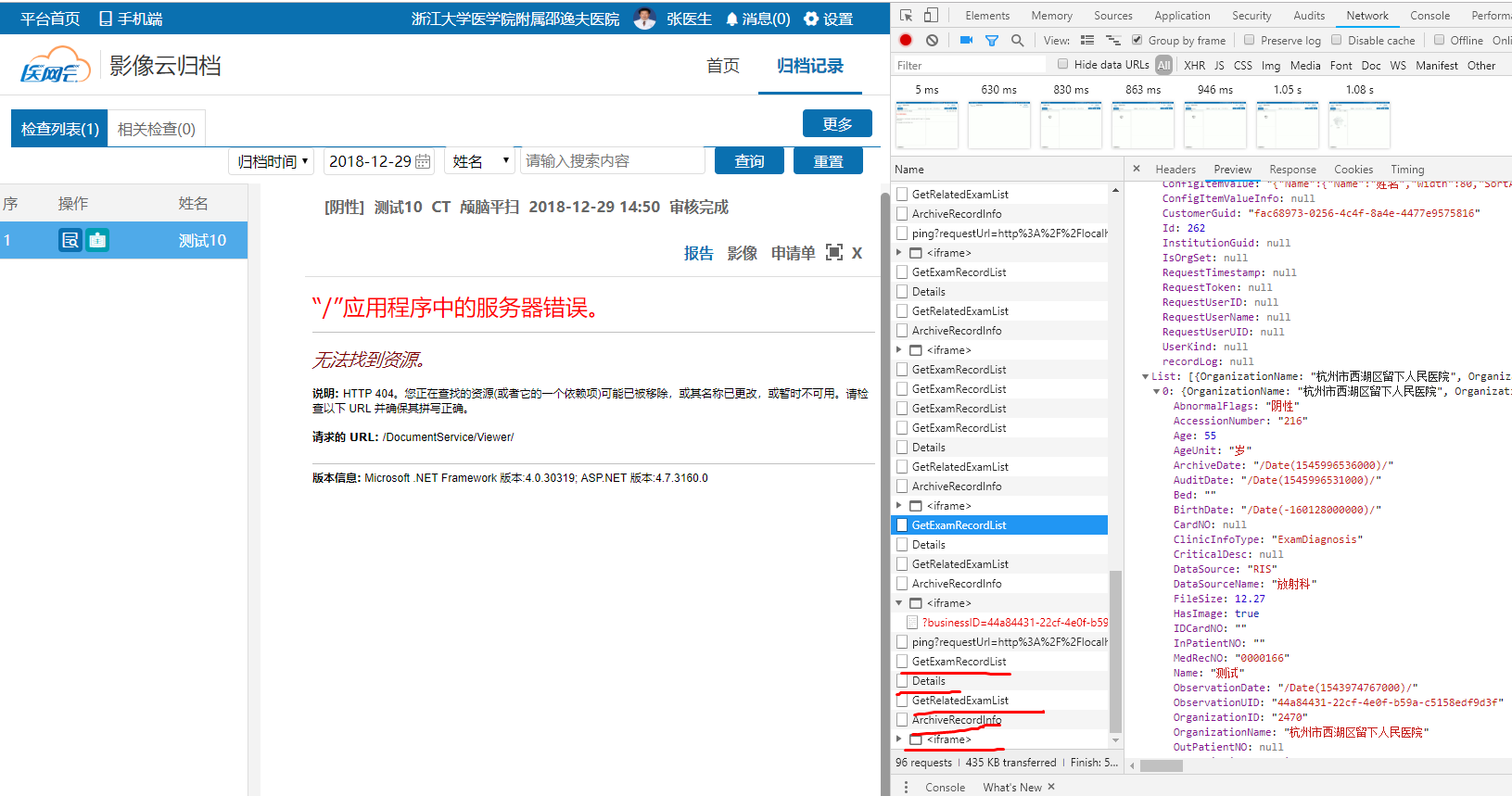
// data.FilmViewUrl = CommonUtil.TransLocalhostForClient(data.FilmViewUrl, url);

//}

//swcout.Stop();

//\_log.Info($"app获取调阅检查详情执行数据4转换，耗时{swcout.ElapsedMilliseconds}毫秒");

7.



发现云归档这边接口所使用的，就是李阳说的方式。应该是陈赵丹这边的第一版。

比方查列表，前端这边请求了5次接口。

（1）检查记录

<http://localhost:9200/SharingVisits/Home/GetExamRecordList>

a. 获取用户配置

//获取用户配置

CustomerConfigInfo customColumnConfig = new CustomerConfigInfo() { ConfigItemName = "share\_exam\_query\_custemfield" };

var customerConfigList = \_crmClient.Invoke(t => t.GetCustomerConfigList(new CustomerConfigQuery()

{

CustomerGuid = \_workContext.CurrentCustomer.CustomerGuid,

ConfigItemName = customColumnConfig.ConfigItemName

}).ResultList);

获取个性化配置的结果：



b. 获取检查记录

根据所有的筛选条件，去ObservationRequest中筛选出所有的满足条件的并且分页的数据。（过程是sql语句的拼接）

也涉及去PatientIndex病人主索引表，还有Visit表中查询数据。

case 0:// 姓名

if (\_patientIndexRepository.Table.Any(x => x.Name == query.CustomQuery && (x.IsMPI != true || x.IsMPI.HasValue == false)))

{

var patientList = \_patientIndexRepository.Table

.Where(x => x.Name == query.CustomQuery && (x.IsMPI != true || x.IsMPI.HasValue == false))

.Select(x => new

{

x.PatientID,

x.PIDAssigningAuthority

})

.ToList();

if (patientList.Count > 0)

{

string querystring = string.Empty;

sqlwhere += " AND (";

foreach (var item in patientList)

{

querystring += " (PatientID='" + item.PatientID + "' AND PIDAssigningAuthority='" + item.PIDAssigningAuthority + "') OR ";

}

querystring = querystring.Substring(0, querystring.LastIndexOf("OR"));

sqlwhere += querystring + " )";

}

}

else

{

result.msg = "未查询到记录";

return result;

}

break;

case 1://病历号

if (\_visitRepository.Table.Any(x => x.MedRecNO == query.CustomQuery))

{

var visitUidList = \_visitRepository.Table

.Where(x => x.MedRecNO == query.CustomQuery)

.Select(x => x.VisitUID).Distinct()

.ToList();

if (visitUidList.Count > 0)

{

string querystring = string.Empty;

sqlwhere += " AND VisitUID IN (";

foreach (var item in visitUidList)

{

querystring += "'" + item + "',";

}

querystring = querystring.Substring(0, querystring.LastIndexOf(","));

sqlwhere += querystring + " )";

}

else

{

result.msg = "未查询到记录";

return result;

}

}

break;

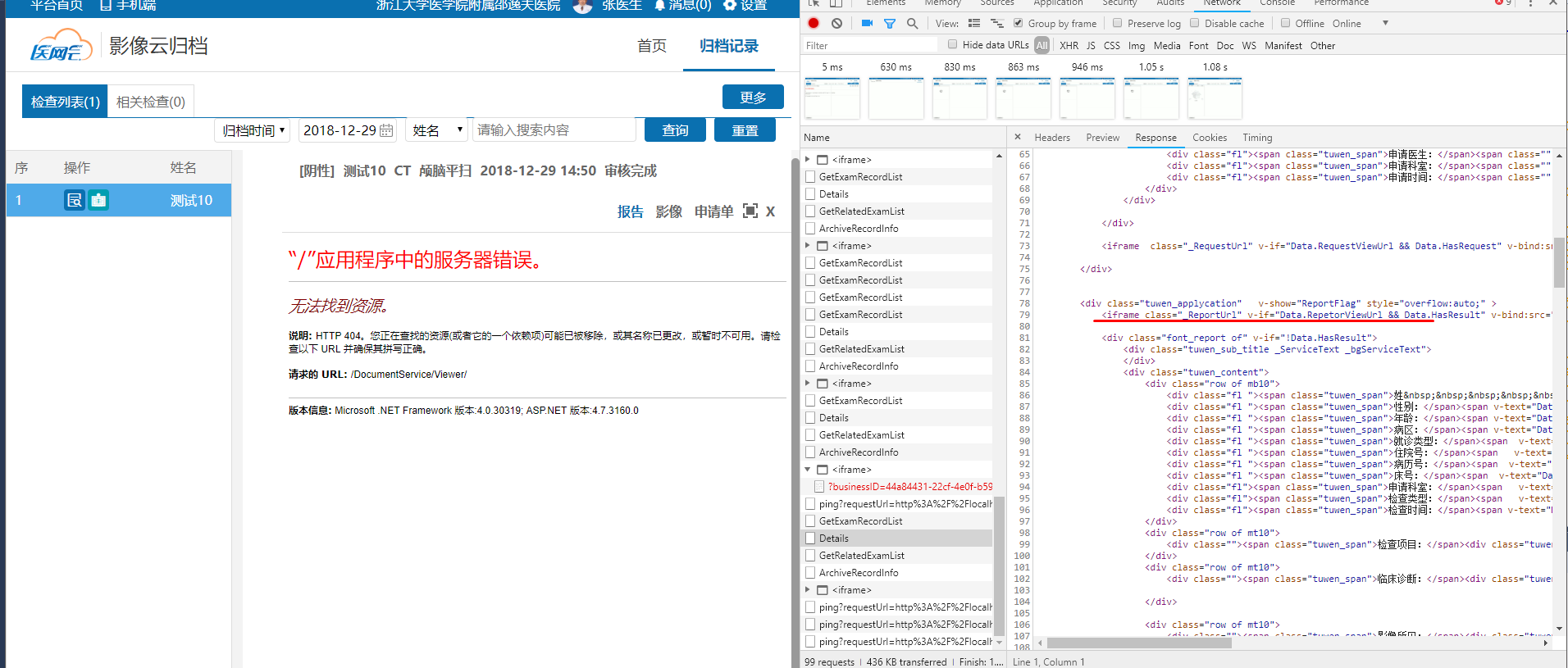
调试最终生成的sql语句（一大堆的数字字符串，表示是在这些机构当中）：

SELECT AccessionNumber, AuditDate, ClinicInfoType, DataSource, ObservationDate, ObservationUID, OrganizationID, OrganizationName, PatientClass, PatientID, PIDAssigningAuthority, PatientMasterID, ProcedureName, ResultDate, ArchiveDate, ServiceSectID, ServiceText, VisitUID,AbnormalFlags,CriticalValue as CriticalDesc,RequestOrgName,RequestDeptName,ProviderName,RegTime,DiskSpace as FileSize,(CASE HasImage WHEN 1 THEN true ELSE false END) as HasImage FROM ObservationRequest WHERE DeleteFlag=0 AND ObservationDate >= '2018/12/29 0:00:00' AND ObservationDate<='2018/12/29 23:59:59' AND OrganizationID IN ('17303','2486','2470','2466','321084100000','2471','2402','100123','100015','100018','88888','18001','18002','18003','18004','18005','18006','1212','17292','2331','2805','15098','17030','47089035133040211A1001','47089034333040211A1001','47002678233040211A1001','PDY10008833040211A1001','47089069333040211A2101','47089053833040211G1001','470970167','33040000001','33048201000','33048200100','33048200500','33048200700','33048200800','33048201500','33048201700','33048201800','33048201100','33048201200','470940099','470940136','470950457','470970263','470970845','470970853','470970888','470970992','731507569','736016734','PDY003181','33040010018','33040020001','33040011002','33041110001','33040000243','33040010041','33040010021','33040010089','33040010033','47106173133048311C1001','47106183833048311A5201','47106189733048311C2201','47106192633048311C1001','47106195033048311C2201','47106207233048311A1001','47106208033048311C2101','47106278533048311C2101','74104230433048311C2201','74202277833048311C2201','PDY10032833048311A7101','jxwjw','33040000001','47100021833042111A2101','47100022633042111A1001','47100023433042111A1001','56819916333042111A5201','47100029333042111G2001','47100003133042111C2201','56937966833042111B1001','47100026933042111C2201','47100224833042111B1001','47100024233042111C2201','47100025033042111C2201','47100013833042111B1001','47100012X33042111C2201','47100016233042111C2201','74051075133041111C1001','74051025X33041111C1001','47095036933040211C1001','txswjj','txsdsrmyy','txszyy','txsdsrmyy','17293','765412','A1200121','9600','1000','15178','17299','001','009','1030001','zsyy1001','gydyrmyy','dzxrmyy','bzqrmyy','dzxlqzwsy','dzxywzzxwsy','bzqpszzxwsy','bzqnbzwsy','150980','17273','17270','dyzfwzz','13198','xdcszdyy','xdcsyy','17308','12938','17302','短信机构','123456123','1','000','111','888','2806','11623','ptqywjj','2440','1234','6833','16036','csjg','987656362','17306','17350','17360','6185','6356','6301','19000','19911','19922','fzjg1','1903454334332','199143543','199453','10001','100011','111101','15176','15098','15099','15102','15107','15120','15118','15110','100002','15105','emdqyy','lszdzx','swlsgkyy','lssgkyy','jwxhhyy','jjlnbzkyy','emdkyy','hjzsgkyy','lssbgtgszkyy','RJYYZY','csfzjg01','TEST1','TEST2','1001','sjftzyyy','mhxzyyy','cs','111','12345','shsdfyyny','jasmpwsw','100001','100007','15246','2314144','1000001','jg493476598','jg1039167575','cswsy','csjcyy','19990915','19980915','19959','19001','19002','19003','19004','123456789','123308244720006231','15649845464','10564964987498','4968956549879456','1222','2313') ORDER BY ObservationDate DESC LIMIT 0, 20;

（2）

<http://localhost:9200/SharingVisits/Home/Details>

返回了当前这个视图，这个页面。



（3）相关检查记录

<http://localhost:9200/SharingVisits/Home/GetRelatedExamList>

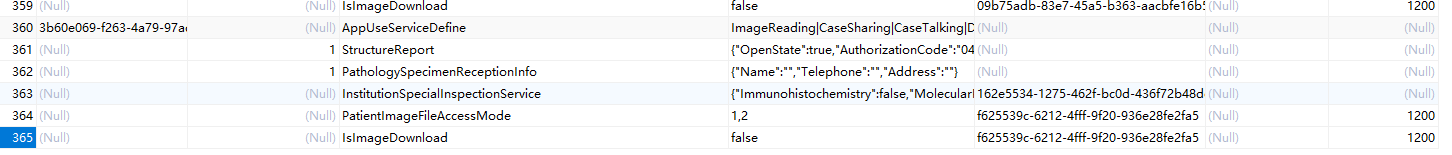
a.

根据条件查询用户配置列表信息

var querySql = from c in \_customerConfigRepository.Table

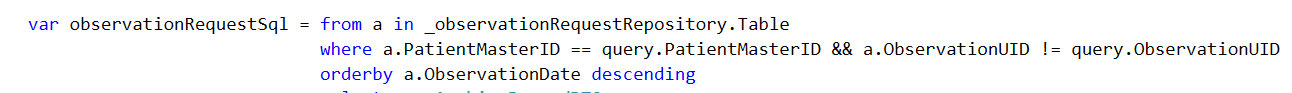
select c;

应该是同一个网页，但每个医生的使用习惯不一样，这里实现了个性化配置。



b. 相关检查列表

针对五张表的操作，先是从ObservationRequest表中查出分页的数据，然后根据病人号，分配机构，病人主索引找到当前病人对应的PatientIndex表，Visit表，DataSouceStep表中的数据，进行组合，返回到前端。



var patientIndexSql = from a in \_patientIndexRepository.Table

where a.PatientMasterID == query.PatientMasterID

select a;

var visitSql = from a in \_visitRepository.Table

where a.PatientMasterID == query.PatientMasterID

select a;

数据来源对应检查类型：

var dataSouceStep = (from a in \_dataSourceSetupRepository.Table

select a).ToList();

（4）获取检查记录详情

<http://localhost:9200/SharingVisits/Home/ArchiveRecordInfo>

a.

通过临床信息分类进入不同的wcf方法

/// <summary>

/// 临床信息分类：2 Lab 检验，1 Exam 检查病人号

/// </summary>

public ClinicInfoTypeEnum ClinicInfoType { get; set; } = ClinicInfoTypeEnum.Exam;

if (query.ClinicInfoType == Core.Enums.ClinicInfoTypeEnum.Lab)

{

var res = \_imageArchiveClient.Invoke(t => t.GetLabArchiveInfo(query));

resultdata.code = res.code;

resultdata.msg = res.msg;

resultdata.Result = res.Result;

}

else

{

var res = \_imageArchiveClient.Invoke(t => t.GetArchiveRecordInfo(query));

resultdata.code = res.code;

resultdata.msg = res.msg;

resultdata.Result = res.Result;

}

第一种：

调用：var archive = GetArchiveInfo(query);

第二种：归档记录详情

var res = \_imageArchiveClient.Invoke(t => t.GetArchiveRecordInfo(query));

调用：var archive = GetArchiveInfo(query);

等于说最终调用的方法还是一样的。

I. 归档记录详情 - 检验

var res = \_imageArchiveClient.Invoke(t => t.GetLabArchiveInfo(query));

（1）先是ObservationRequest中获取数据

var observationRequest = \_observationRequestRepository.Table.FirstOrDefault(x => x.ObservationUID == uid && x.DeleteFlag == false);

（2）匹配主索引表获取病人信息

var patient = \_patientIndexRepository.Table.FirstOrDefault(x => x.PatientMasterID == observationRequest.PatientMasterID && x.PatientID == observationRequest.PatientID && x.PIDAssigningAuthority == observationRequest.PIDAssigningAuthority);

（3）数据来源对应检查类型

var dataSouceStep = \_dataSourceSetupRepository.Table.FirstOrDefault(a => a.DataSource == observationRequest.DataSource);

（4）从ObservationResult中获取两条检查结果。

var observationResultList = \_observationResultRepository.Table.Where(x => x.ObservationUID == observationRequest.ObservationUID)

.Select(x => new { x.ValueText, x.ObservationID }).ToList();

（5）根据是否是启用短链接的判断，进入到不同的获取文档的方法。

var documentList = new Core.BasicResult.ModelResult<List<DocumentViewDTO>>();

if (ConfigDefine.UseShortDocumentViewerUrl)

{

documentList = \_documentService.GetPatientExamDocumentList(new DocumentQuery

{

BusinessID = businessid,

BusinessType = archiveInfo.ClinicInfoType,

OrganizationID = observationRequest.OrganizationID,

WithEWorldViewerUrl = true

});

}

else

{

documentList = \_documentService.GetExamDocumentList(new DocumentQuery

{

BusinessID = businessid,

BusinessType = archiveInfo.ClinicInfoType,

OrganizationID = observationRequest.OrganizationID,

WithEWorldViewerUrl = true

});

}

第一种：GetPatientExamDocumentList：

根据OrganizationID去配置表中获取内外网地址等配置。



var service = new ServiceConfigInfo();

var list = \_serviceConfigService.GetServiceConfigList(new ServiceConfigQuery()

{

ServiceType = ServiceTypeEnum.DocumentService,

CurrentFlag = true

}).ResultList;

if (list.Any())

{

if (!string.IsNullOrWhiteSpace(OrganizationID))

{

if (list.Any(a => a.OrganizationID == OrganizationID))

{

list = list.Where(a => a.OrganizationID == OrganizationID);

}

else

{

list = list.Where(a => a.OrganizationID == null);

}

}

else

{

list = list.Where(a => a.OrganizationID == null);

}

service = list.FirstOrDefault();

if (service != null)

{

var internetEndpoint = service.UseTLS == false ? service.EndpointInternet : service.EndpointTLSInternet;

if (!string.IsNullOrWhiteSpace(internetEndpoint))

{

\_log.Debug("检测到配置表中有相应外网地址，外网地址=" + internetEndpoint);

result.InternetEndpoint = internetEndpoint;

}

var intranetEndpoint = service.UseTLS == false ? service.Endpoint : service.EndpointTLS;

if (!string.IsNullOrEmpty(intranetEndpoint))

{

\_log.Debug("检测到配置表中有相应内网地址，内网地址=" + intranetEndpoint);

result.IntranetEndpoint = intranetEndpoint;

}

if (string.IsNullOrWhiteSpace(internetEndpoint) && string.IsNullOrWhiteSpace(intranetEndpoint))

{

\_log.Debug("未检测到相应外网地址和内网地址，使用web服务地址作为内网地址。");

result.IntranetEndpoint = GetDocumentServiceUrl();

return result;

}

else

{

\_log.Error("外网地址，内网地址，网站地址都没找到或不可用。");

return result;

}

}

else

{

if (!string.IsNullOrWhiteSpace(ConfigDefine.IDCASUrl))

{

result.IntranetEndpoint = GetDocumentServiceUrl();

}

else

{

\_log.Error("外网地址，内网地址，网站地址都没找到或不可用。");

}

return result;

}

}

else

{

\_log.Debug("未检测到文档服务配置，开始获取当前Web地址。");

if (!string.IsNullOrWhiteSpace(ConfigDefine.IDCASUrl))

{

result.IntranetEndpoint = GetDocumentServiceUrl();

return result;

}

else

{

\_log.Error("外网地址，内网地址，网站地址都没找到或不可用。");

return result;

}

}

获取配置之后对url地址进行拼接，组装：

if (shortModel.code == ResultCodeEnum.Success)

{

model.FileBusinessViewUrl = endpoint + "/Viewer/ShortUrl/" + shortModel.Result;

model.FileViewUrl = endpoint + "/Viewer/ShortUrl/" + shortModel.Result;

}

#region withDicomDownload或者withDicomdirPath

if (query.WithDicomDownload || query.WithDicomdirPath)

{

if (!string.IsNullOrWhiteSpace(endpoint))

{

var webclient = new WebClientEx();

webclient.Headers.Add("Content-Type", "application/x-www-form-urlencoded");

if (item.FormatCode == "DICOMDIR")

{

try

{

var getDICOMDownloadListUrl = endpoint +

"/Viewer/GetDICOMDownloadList?fileUid=" + item.FileUID + "&token=" +

CommonUtil.CreateWCFToken(item.FileUID.ToString()) +

"&withDicomDownload=" + query.WithDicomDownload;

\_log.Info(string.Format("DICOMDownloadListUrl={0}", getDICOMDownloadListUrl));

var data = webclient.UploadData(getDICOMDownloadListUrl, "POST", new byte[0]);

var dateStr = HttpUtility.UrlDecode(Encoding.ASCII.GetString(data));

var dicomdirDownload = JsonHelper.DeserializeFromJsonString<DicomdirDownloadInfo>(dateStr);

model.DICOMDownloadList = dicomdirDownload.DcmList;

}

catch (Exception e)

{

\_log.Error("生成dicom文件下载地址失败，源dicom文件不存在", e);

continue;

}

}

}

}

第二种：GetExamDocumentList：

总之也是url地址的拼接：

if (documentList.Count > 0)

{

List<DocumentViewDTO> resultList = null;

#region 数据处理

resultList = documentList.Select(d => new DocumentViewDTO

{

FileUID = d.FileUID,

MimeType = d.MimeType,

BusinessID = d.BusinessID,

BusinessType = d.BusinessType,

ClassCode = d.ClassCode,

TypeCode = d.TypeCode,

FormatCode = d.FormatCode,

FileSHA = d.FileSHA,

FileSize = d.FileSize,

UniqueID = d.UniqueID,

FileCreateTime = d.FileCreateTime,

OriginalFileName = d.OriginalFileName,

FileDownloadUrl = endpoint + "/Download/Document/?fileUid=" + d.FileUID +

"&token=" + CommonUtil.CreateWCFToken(d.FileUID.ToString()),

FileViewUrl = endpoint + "/Viewer/?businessID=" + d.BusinessID +

"&businessType=" + d.BusinessType + "&classCode=" + d.ClassCode +

"&typeCode=" + d.TypeCode + "&fileUid=" + d.FileUID +

"&organizationID=" + (string.IsNullOrWhiteSpace(query.OrganizationID) ? d.CreateOrganizationID : query.OrganizationID) +

"&print=" + query.Print + "&download=" + query.Download + "&businessView=false&token=" +

CommonUtil.CreateWCFToken(d.BusinessID + d.BusinessType + d.ClassCode + d.TypeCode +

d.FileUID + "00") + "&reportimgpost=" + query.reportImgPost + "&Anonymous=" + query.Anonymous,

FileBusinessViewUrl = endpoint + "/Viewer/?businessID=" + d.BusinessID +

"&businessType=" + d.BusinessType + "&classCode=" + d.ClassCode +

"&typeCode=" + d.TypeCode + "&fileUid=" + d.FileUID +

"&organizationID=" + (string.IsNullOrWhiteSpace(query.OrganizationID) ? d.CreateOrganizationID : query.OrganizationID) +

"&print=" + query.Print + "&download=" + query.Download +

"&businessView=true&token=" +

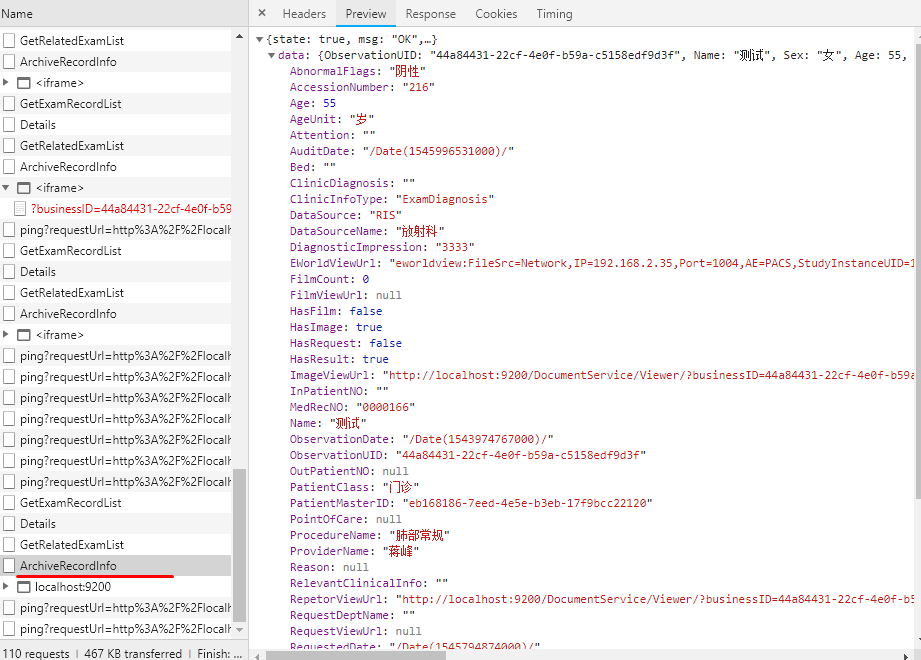
CommonUtil.CreateWCFToken(d.BusinessID + d.BusinessType + d.ClassCode +

d.TypeCode + d.FileUID + "00") + "&reportimgpost=" + query.reportImgPost + "&Anonymous=" + query.Anonymous

}).ToList();

（6）暂时没看到Dicom相关的操作。

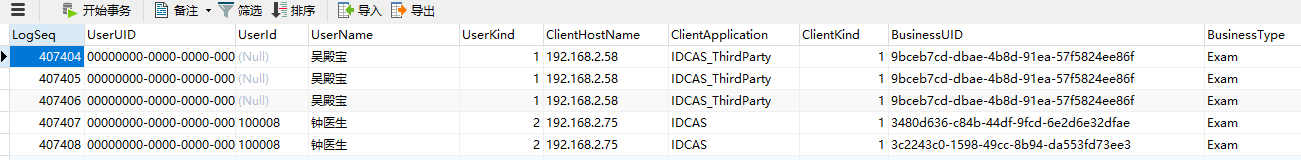
结果就是返回了检查记录的详情：



（5）添加临床操作日志

\_imageArchiveClient.Invoke(i => i.AddClinicDataLog(addLog));

\_logClinicDataOperationRepository.Insert(log);



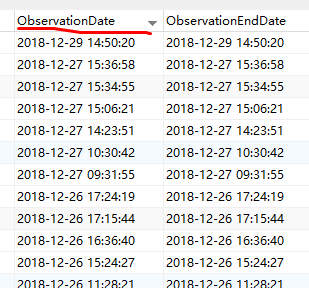
调试发现，选择检查时间，没有数据。



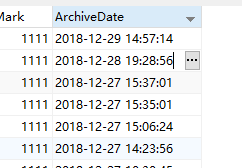
但换成归档时间，就有数据了。



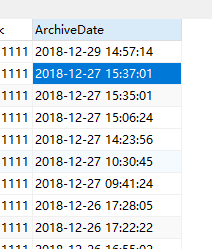
对的，观察时间，28号没数据。



但是归档时间，28号有数据的。



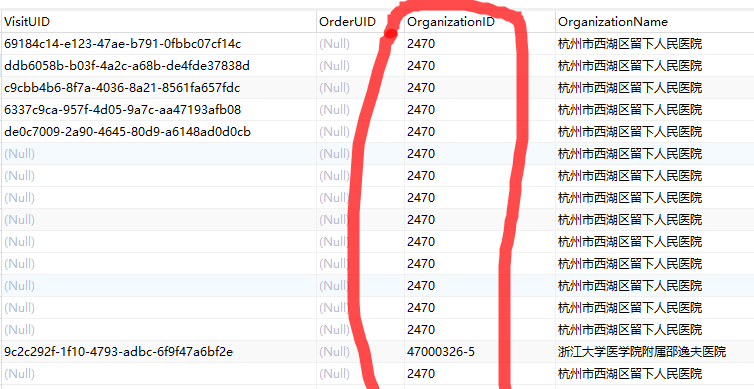
但是27号归档，数据库中是有数据的：



但是调试查询，时间也是对的：

" AND ArchiveDate>= '2018/12/27 0:00:00' AND ArchiveDate<='2018/12/27 23:59:59'"

而且机构2470也是包含在sql语句中的。



最后生成的sql语句：

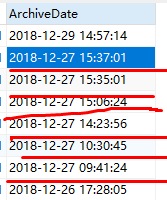
" SELECT AccessionNumber, AuditDate, ClinicInfoType, DataSource, ObservationDate, ObservationUID, OrganizationID, OrganizationName, PatientClass, PatientID, PIDAssigningAuthority, PatientMasterID, ProcedureName, ResultDate, ArchiveDate, ServiceSectID, ServiceText, VisitUID,AbnormalFlags,CriticalValue as CriticalDesc,RequestOrgName,RequestDeptName,ProviderName,RegTime,DiskSpace as FileSize,(CASE HasImage WHEN 1 THEN true ELSE false END) as HasImage FROM ObservationRequest WHERE DeleteFlag=0 AND ArchiveDate>= '2018/12/27 0:00:00' AND ArchiveDate<='2018/12/27 23:59:59' AND OrganizationID IN ('17303','2486','2470','2466','321084100000','2471','2402','100123','100015','100018','88888','18001','18002','18003','18004','18005','18006','1212','17292','2331','2805','15098','17030','47089035133040211A1001','47089034333040211A1001','47002678233040211A1001','PDY10008833040211A1001','47089069333040211A2101','47089053833040211G1001','470970167','33040000001','33048201000','33048200100','33048200500','33048200700','33048200800','33048201500','33048201700','33048201800','33048201100','33048201200','470940099','470940136','470950457','470970263','470970845','470970853','470970888','470970992','731507569','736016734','PDY003181','33040010018','33040020001','33040011002','33041110001','33040000243','33040010041','33040010021','33040010089','33040010033','47106173133048311C1001','47106183833048311A5201','47106189733048311C2201','47106192633048311C1001','47106195033048311C2201','47106207233048311A1001','47106208033048311C2101','47106278533048311C2101','74104230433048311C2201','74202277833048311C2201','PDY10032833048311A7101','jxwjw','33040000001','47100021833042111A2101','47100022633042111A1001','47100023433042111A1001','56819916333042111A5201','47100029333042111G2001','47100003133042111C2201','56937966833042111B1001','47100026933042111C2201','47100224833042111B1001','47100024233042111C2201','47100025033042111C2201','47100013833042111B1001','47100012X33042111C2201','47100016233042111C2201','74051075133041111C1001','74051025X33041111C1001','47095036933040211C1001','txswjj','txsdsrmyy','txszyy','txsdsrmyy','17293','765412','A1200121','9600','1000','15178','17299','001','009','1030001','zsyy1001','gydyrmyy','dzxrmyy','bzqrmyy','dzxlqzwsy','dzxywzzxwsy','bzqpszzxwsy','bzqnbzwsy','150980','17273','17270','dyzfwzz','13198','xdcszdyy','xdcsyy','17308','12938','17302','短信机构','123456123','1','000','111','888','2806','11623','ptqywjj','2440','1234','6833','16036','csjg','987656362','17306','17350','17360','6185','6356','6301','19000','19911','19922','fzjg1','1903454334332','199143543','199453','10001','100011','111101','15176','15098','15099','15102','15107','15120','15118','15110','100002','15105','emdqyy','lszdzx','swlsgkyy','lssgkyy','jwxhhyy','jjlnbzkyy','emdkyy','hjzsgkyy','lssbgtgszkyy','RJYYZY','csfzjg01','TEST1','TEST2','1001','sjftzyyy','mhxzyyy','cs','111','12345','shsdfyyny','jasmpwsw','100001','100007','15246','2314144','1000001','jg493476598','jg1039167575','cswsy','csjcyy','19990915','19980915','19959','19001','19002','19003','19004','123456789','123308244720006231','15649845464','10564964987498','4968956549879456','1222','2313') ORDER BY ObservationDate DESC LIMIT 0, 20; "

查询数量count：

" SELECT COUNT(1) from ( SELECT AccessionNumber, AuditDate, ClinicInfoType, DataSource, ObservationDate, ObservationUID, OrganizationID, OrganizationName, PatientClass, PatientID,PIDAssigningAuthority, PatientMasterID, ProcedureName, ResultDate, ArchiveDate, ServiceSectID, ServiceText, VisitUID,AbnormalFlags,CriticalValue as CriticalDesc,RequestOrgName,RequestDeptName,ProviderName,RegTime,DiskSpace as FileSize,(CASE HasImage WHEN 1 THEN 1 ELSE 0 END) as HasImage FROM ObservationRequest WHERE DeleteFlag=0 AND ArchiveDate>= '2018/12/27 0:00:00' AND ArchiveDate<='2018/12/27 23:59:59' AND OrganizationID IN ('17303','2486','2470','2466','321084100000','2471','2402','100123','100015','100018','88888','18001','18002','18003','18004','18005','18006','1212','17292','2331','2805','15098','17030','47089035133040211A1001','47089034333040211A1001','47002678233040211A1001','PDY10008833040211A1001','47089069333040211A2101','47089053833040211G1001','470970167','33040000001','33048201000','33048200100','33048200500','33048200700','33048200800','33048201500','33048201700','33048201800','33048201100','33048201200','470940099','470940136','470950457','470970263','470970845','470970853','470970888','470970992','731507569','736016734','PDY003181','33040010018','33040020001','33040011002','33041110001','33040000243','33040010041','33040010021','33040010089','33040010033','47106173133048311C1001','47106183833048311A5201','47106189733048311C2201','47106192633048311C1001','47106195033048311C2201','47106207233048311A1001','47106208033048311C2101','47106278533048311C2101','74104230433048311C2201','74202277833048311C2201','PDY10032833048311A7101','jxwjw','33040000001','47100021833042111A2101','47100022633042111A1001','47100023433042111A1001','56819916333042111A5201','47100029333042111G2001','47100003133042111C2201','56937966833042111B1001','47100026933042111C2201','47100224833042111B1001','47100024233042111C2201','47100025033042111C2201','47100013833042111B1001','47100012X33042111C2201','47100016233042111C2201','74051075133041111C1001','74051025X33041111C1001','47095036933040211C1001','txswjj','txsdsrmyy','txszyy','txsdsrmyy','17293','765412','A1200121','9600','1000','15178','17299','001','009','1030001','zsyy1001','gydyrmyy','dzxrmyy','bzqrmyy','dzxlqzwsy','dzxywzzxwsy','bzqpszzxwsy','bzqnbzwsy','150980','17273','17270','dyzfwzz','13198','xdcszdyy','xdcsyy','17308','12938','17302','短信机构','123456123','1','000','111','888','2806','11623','ptqywjj','2440','1234','6833','16036','csjg','987656362','17306','17350','17360','6185','6356','6301','19000','19911','19922','fzjg1','1903454334332','199143543','199453','10001','100011','111101','15176','15098','15099','15102','15107','15120','15118','15110','100002','15105','emdqyy','lszdzx','swlsgkyy','lssgkyy','jwxhhyy','jjlnbzkyy','emdkyy','hjzsgkyy','lssbgtgszkyy','RJYYZY','csfzjg01','TEST1','TEST2','1001','sjftzyyy','mhxzyyy','cs','111','12345','shsdfyyny','jasmpwsw','100001','100007','15246','2314144','1000001','jg493476598','jg1039167575','cswsy','csjcyy','19990915','19980915','19959','19001','19002','19003','19004','123456789','123308244720006231','15649845464','10564964987498','4968956549879456','1222','2313')) t"

数量的结果是对的：





执行查询ObservationRequest表的数据时报错了。

list = \_observationRequestRepository.SqlQuery<ArchiveRecordDTO>(sql).ToList();

原来是报错了：



{"The cast to value type 'System.Guid' failed because the materialized value is null. Either the result type's generic parameter or the query must use a nullable type."}

{“转换为值类型'System.Guid'失败，因为具体化值为null。结果类型的泛型参数或查询必须使用可空类型。”}

感觉这类的解决方法，把guid变成guid?就可以了。

试试看。

/// <summary>

/// 检查业务唯一号

/// </summary>

public Guid? ObservationUID { get; set; }

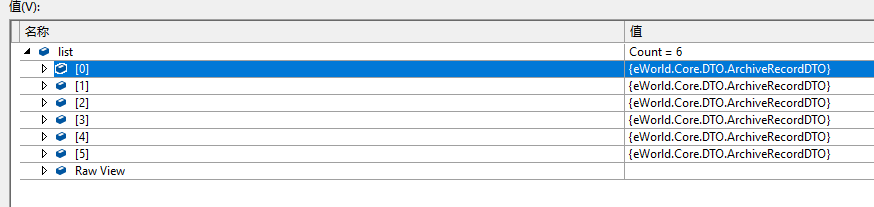
/// <summary>

/// 检查申请UID

/// </summary>

public Guid? VisitUID { get; set; }

果然就好了。解决了一个问题，很好。



页面有数据了。



还有，总感觉，这个花花，是我在协同的时候提交上来的。

8.

9.

10.

11.

12.

13.