护理记录按规则配置查看天数

1. 需求

护理记录在不同的科室由于患者住院天数的长短不一致，为了显示性能，有时需要展示住院期间某一段时间的护理记录，有时需要展示全部护理记录。

通过启用护理记录查看天数规则计算，能够根据患者、科室、病区、模板通过规则返回不同的查看天数，对应转换为开始时间。

1. 配置说明

根据护理记录管理的需要，将护理记录的管控分为患者级和模板级级，同时提供模板的排序管理功能

患者级管理，只管控每位患者能够查看文书模板的时间范围；

模板级管理，即可以定制模板的允许查看时间范围及定制专用属性；

排序管理，在规则中将要使用的模板按顺序排序，一旦启用，就需要将要展示的模板均放置进来，否则只会显示排序管理中返回的模板

患者级管理和模板级管理先进行患者级管理规则执行，再执行模板级管理规则，后面的规则覆盖前面的规则，如果只想启用患者级管理，只需将模板级规则的条件设置为“1” == “2”，使用模板级规则返回为空即可。

1. 配置操作步骤
	1. 患者级管理
2. 启用天数计算规则

在系统管理->配置管理->参数配置中添加添加：name：通过规则返回护理记录查看天数，key：GET\_HL\_RECORD\_VIEWRANGE\_BY\_RULE，值：yes

1. 配置数据源

如果传入参数无法完成规则处理，需要结合外部数据时，可以先配置好数据源，在规则中调用数据源返回结果后再处理。

本例配置中需要将护理记录的查看开始时间从患者住院天开始，在系统管理->数据管理->数据源配置中维护数据源，点击新增进入数据维护界面，填写名称和查询脚本，支持sql和存储过程两种方式，根据页面提示维护即可。



这里通过sql脚本来处理，对应sql脚本为：

select pe.patient\_encounter\_id as ENC\_ID, TO\_DATE(TO\_CHAR(sysdate,'YYYY-MM-DD'),'YYYY-MM-DD') - TO\_DATE(TO\_CHAR(nvl(pe.effective\_b, pe.activity\_b),'YYYY-MM-DD'),'YYYY-MM-DD') as datascope from patient\_encounter pe

在代码段中维护条件： ENC\_ID = '${ENC\_ID}' ，此处无须添加where条件，最后程序处理时会将上面两段脚本组装成：Select temp.\* from (\_sql\_) as temp where 1=1 and ENC\_ID = '${ENC\_ID}' 样式，会自动添加进行\_sql\_脚本替换，这就要求代码段中的字段只能取\_sql\_中的返回字段。${ENC\_ID}为参数点位符，格式固定为${ }形式，ENC\_ID执行数据源时传入的参数。

配置完成后保存，生成返回列，点击测试进行数据源测试：

{"ENC\_ID":"2300000454"}



测试结果：



此时表示数据源执行成功；

1. 配置天数计算规则

在系统管理->配置管理->规则配置中添加规则：规则标识：HL\_RECORD\_VIEWRANGE\_RULE 规则描述：护理记录可查看天数，根据科室、病区、模板调用规则返回。规则标识固定不能允许更改，规则描述自由描述即可。



点击自定义规则进行规则维护界面：



脚本如下:

package cnis.doc;

import java.lang.\*;

import java.util.\*;

import com.pkuhit.cnis.core.utils.SpringUtil;

import com.pkuhit.cnis.base.dataSourceManager.service.DataSourceService

global java.util.HashMap result;

//下面内容为可编辑区 begin

rule "dataRange\_1"

salience 1

when

 $map:Map(((String)this["encId"]) != "") // encId为外部传入参数,判断encId不为空

Then

 //获取encId和templateCode

 String encId = (String) $map.get("encId");

 String templateCode = (String) $map.get("templateCode");

 //调用方法返回住院天数

 Integer admissDay = returnAdmissDay(encId);

 //判断住院天数是否小于0，小于0时默认返回3天，可修改

 if(admissDay <= 0){

 admissDay = 3;

 }

 //定义一个map对象，将值装入后返回

 Map<String,Integer> retMap = new HashMap();

 retMap.put("viewRange",admissDay);

 //将结果放到result中，result作为返回对象最后会将结果带回

 result.put("result",retMap);

end

//到此为一个完整的规则，向下可复制多个，根据条件会顺序执行

//下面是获取住院天数的方法，参数可传入多个，在调用时按顺序写入即可

function Integer returnAdmissDay(String encId){

 // 数据源编码，需要数据源管理中维护完成后查询得出select \* from DATA\_SOURCE\_MANAGER where //nm='住院天数'

 String dataSourceCode = "47fd570537c345e7a99acef462b8afb8";

 // 数据源参数，将encId值作为参数传入数据源执行器

 Map<String,Map<String,Object>> param = new HashMap<String,Map<String,Object>>();

 Map<String,Object> data = new HashMap<String,Object>();

 data.put("ENC\_ID",encId);

 //组装上数据源编码和参数

 param.put(dataSourceCode,data);

 // 执行数据源

 DataSourceService dataSourceService = (DataSourceService)SpringUtil.getBean( DataSourceService.class);

 List<Map> datasourceCodeList = dataSourceService.execDataSource(param);

 //获取返回结果

 List<Map<String, Object>> sqlResult = new ArrayList<>();

 if (datasourceCodeList != null && datasourceCodeList.size() > 0) {

 sqlResult = (List<Map<String, Object>>) datasourceCodeList.get(0).get(dataSourceCode);

 }

 // 数据源执行结果

 Map<String,Object> res = new HashMap<String,Object>();

 if(sqlResult.size() > 0){

 res = sqlResult.get(0);

 }

 // 返回数据源执行结果 测试直接返回结果

 Integer admissDay = Integer.parseInt((res.get("DATASCOPE")==null?"0":res.get("DATASCOPE").toString()));

 return admissDay;

}

中间规则部分可复制多个。保存规则后，点击测试传入参数进行调用测试：参数格式为{“key1”:”value1”,”key2”:”value2”}，多个参数顺序添加即可。

本例中需要会传入encId和templateCode，则组装为：

{"encId":"2300000454","templateCode":"123"}

点击测试查看执行结果：



执行返回结果正确则表示规则配置正确，可在程序中进行使用。

1. 效果展示





根据规则，开始时间为住院数天，不同的患者进入页面后，对应的开始时间均为当前患者的住院开始时间，

* 1. 模板级管理
1. 规则定义

在系统管理->配置管理->规则管理中添加规则：HL\_RECORD\_ADDRULE，描述信息：护理记录可查看天数范围(可自由修改)，保存后，点击【自定义规则】进入规则维护界面：



脚本如下：

package cnis.rule;

import java.lang.\*;

import java.util.\*;

import com.pkuhit.cnis.core.utils.SpringUtil;

import com.pkuhit.cnis.base.dataSourceManager.service.DataSourceService;

global java.util.HashMap result;

rule "hl\_record\_add"

salience 1

when

 $map:Map( "1" == "1" ) //“1”==“2”时暂不启用

Then

 //定义一个返回对象

Map item = new HashMap();

//模板编码，从nur\_deo,通过select \* from NUR\_DOC\_TEMPLET where nur\_doc\_temp\_name like '%记录单%' //从文书模板中获取nur\_doc\_temp\_code即可

item.put("templateCode","2cd331e833d04cc38518636ab095d177");

//开始时间，这里7表示当前时间减去7天

item.put("showStartDate", "7");

//结束时间

item.put("showEndDate", "0");

//将对象放对返回结果集

 result.put(1,item);

 //下面的逻辑同上，有多少个模板复制添加修改即可

 Map item1 = new HashMap();

 item1.put("templateCode","9adf46374c1f4a949b9c21c570fbeb68");

 item1.put("showStartDate", "7");

 item1.put("showEndDate", "0");

 result.put(2,item1);

 Map item2 = new HashMap();

 item2.put("templateCode","70de4de16fc64a2eb3a3918bf5d99970");

 item2.put("showStartDate", "7");

 item2.put("showEndDate", "0");

result.put(3,item2);

end

保存后执行测试：



1. 效果展示

设置显示7天：



改成14天后：



如果只想按患者来管理，可以将模板管理的规则设置为不启用：



此时，护理记录的开始时间就变成患者的入院时间了：



* 1. 模板排序
1. 规则定义

在系统管理->配置管理->规则管理下新增规则：HLRECORDSORTRULE，规则描述自由定义，例如：护理记录排序计算规则，一旦启用(记录数大于0)返回护理记录单列表以此规则列表为准，保存后点击【自定义规则】进行规则配置：



脚本如下：

package cnis.rule;

import java.lang.\*;

import java.util.\*;

import com.pkuhit.cnis.core.utils.SpringUtil;

import com.pkuhit.cnis.base.dataSourceManager.service.DataSourceService;

global java.util.HashMap result;

rule "hl\_record\_sort"

salience 1

when

 $map:Map( "1" == "1" )

then

Map item = new HashMap();

//护理记录单

 item.put("templateCode","9a63dff1d0524c498a8caeb18a279cf6");

 result.put(1,item);

Map item1 = new HashMap();

//格拉斯哥昏迷评分（GCS

item1.put("templateCode","95432647db654dbfa10c66ed38918e4e"); result.put(2,item1);

Map item2 = new HashMap();

//交接班记录单(手术科室)

item2.put("templateCode","b789334a3b9c4e4c8c8c3145ad8e4843"); result.put(3,item2);

end

需要调整前台的显示顺序时，只要换上述各个模板的位置即可。

1. 效果展示



调整规则中的模板顺序，调整put的key值1，2，3对应的记录单即可：



前面展示结果：



注释掉其中一个模板后：



前端只展示剩下的2个模板



一旦启用排序规则后，如果有新增的记录单要显示，需要添加到规则中。

1. 总结

根据医院需求，护理记录的默认时间周期，模板顺序，以及显示哪些模板都可以通过规则做进一步的细化处理。