# 机台任务下发

JSONObject 组成：按工单内容发送

PlaCourse 一个工单

List<PlaMachinePlan> 机台计划列表

PlaMachinePlan 单个机台计划

List< PlaMachinePlanMater> 原料数量列表

CraCraftProduct 产品工艺

CraRoute 产品工艺路线

List<CraRouteSeq> 对应的工序列表

CraRouteSeq 单个工序

List< CraSeqParam> 各工序对应的需收集参数列表

**PlaCourse对象：**

@Id

@Column(name="id")

@GeneratedValue(strategy = GenerationType.*IDENTITY*)

**private** Integer id;

@Column(name = "create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date createDate;

@Column(name="create\_by")

**private** String createBy;

@Column(name="remark")

**private** String remark;

@Column(name="ws\_type")

**private** String wsType;

@Column(name="ws\_code")

**private** String wsCode;

@Column(name="sc\_code")

**private** String scCode;

@Column(name="bat\_code")

**private** String batCode;

@Column(name="head\_ggxh")

**private** String headGgxh;

@Column(name = "color")

**private** String color;

@Column(name = "head\_zzdc")

**private** String headZzdc;

@Column(name = "head\_zzds")

**private** String headZzds;

@Column(name = "total\_amount")

**private** Integer totalAmount;

@Column(name = "manu\_notice\_id")

**private** Integer manuNoticeId;

@Column(name = "cus\_id")

**private** String cusId;

@Column(name = "bill\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date billDate;

**//开单日期 重要**

**@Transient**

**private Long billDate\_Long; //新增**

@Column(name = "is\_finish")

**private** String isFinish;

@Column(name = "c\_code")

**private** String cCode;

@Column(name = "audit\_flag")

**private** String auditFlag;

@Column(name = "audit\_time")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date auditTime;

@Column(name = "course\_name")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date courseName;

@Column(name = "use\_flag")

**private** String useFlag;

@Column(name = "manu\_tenet")

**private** String manuTenet;

@Column(name = "past\_flag")

**private** String pastFlag;

@Column(name = "plan\_enable\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date planEnableDate;

@Column(name = "route\_code")

**private** String routeCode;

@Column(name = "pro\_craft\_code")

**private** String proCraftCode;

//工单的机台计划

@Transient

**private** List<PlaMachinePlan> planLst = **new** ArrayList<PlaMachinePlan>();

//计划材料

@Transient

**private** List<PlaMachinePlanMater> materList = **new** ArrayList<PlaMachinePlanMater>();

//产品工艺。内包含各工序的工艺参数

@Transient

**private** CraCraftProduct craft = **new** CraCraftProduct();

//机台名称  
@Transient  
private String machineName; //新增

**PlaMachinePlan对象：**

@Id

@Column(name = "id")

**private** Integer id;

@Column(name = "week\_plan\_id")

**private** Integer weekPlanId;

@Column(name = "work\_date")

//@Temporal(TemporalType.TIMESTAMP)

@DateTimeFormat(pattern = "yyyy-MM-dd")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Timestamp workDate;

@Column(name = "machine\_id")

**private** Integer machineId;

@Column(name = "employee\_id")

**private** Integer employeeId;

@Column(name = "axis\_name")

**private** String axisName;

@Column(name = "plan\_start\_time")

//@Temporal(TemporalType.TIMESTAMP)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date planStartTime;

@Column(name = "plan\_end\_time")

//@Temporal(TemporalType.TIMESTAMP)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Timestamp planEndTime;

@Column(name = "fact\_start\_time")

//@Temporal(TemporalType.TIMESTAMP)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Timestamp factStartTime;

@Column(name = "fact\_end\_time")

//@Temporal(TemporalType.TIMESTAMP)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Timestamp factEndTime;

@Column(name = "end\_axis\_code")

**private** String endAxisCode;

@Column(name = "route\_code")

**private** String routeCode;

@Column(name = "part\_len")

**private** String partLen;

@Column(name = "product\_state")

**private** String productState;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date createDate;

@Column(name = "course\_code")

**private** String courseCode;

**//下面新增**

**//工作日期**

**@Transient**

**private Long workDate\_inLong;**

**//计划开工时间**

**@Transient**

**private Long planStartTime\_inLong;**

**//计划结束时间**

**@Transient**

**private Long planEndTime\_inLong;**

//计划材料

@Transient

**private** List<PlaMachinePlanMater> materList = **new** ArrayList<PlaMachinePlanMater>();

**PlaMachinePlanMater对象：**

@Id

@Column(name = "id")

//@GeneratedValue(strategy = GenerationType.IDENTITY)

**private** Integer id;

@Column(name = "machine\_plan\_id")

**private** Integer machinePlanId;

@Column(name = "mater\_id")

**private** Integer materId;

@Column(name = "amount")

**private** Double amount;

@Column(name = "unit")

**private** String unit;

@Column(name = "ggxh")

**private** String ggxh;

@Column(name = "axis\_name")

**private** String axisName;

@Column(name = "machine\_id")

**private** Integer machineId;

@Column(name = "course\_code")

**private** String courseCode;

@Column(name = "plan\_send\_time")

**private** Timestamp planSendTime;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

**private** Date createDate;

@Column(name = "flag")

**private** String flag;

@Column(name = "send\_state")

**private** String sendState;

@Column(name = "mater\_name")

**private** String materName;

**//新增**

**//原料计划配送时间**

**@Transient**

**private Long planSendTime\_inLong;**

**CraCraftProduct产品工艺对象：**

@Id

@Column(name = "id")

**private** Integer id;

@Column(name = "pro\_craft\_code")

**private** String proCraftCode;

@Column(name = "pro\_craft\_name")

**private** String proCraftName;

@Column(name = "pro\_gg")

**private** String proGg;

@Column(name = "pro\_xh")

**private** String proXh;

@Column(name = "pro\_id")

**private** Integer proId;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date createDate;

@Column(name = "route\_code")

**private** String routeCode;

@Column(name = "pro\_ggxh")

**private** String proGgxh;

@Column(name = "core\_account")

**private** Integer coreAccount;

/\*\*

\* 产品工艺 + 工艺路线->工序列表->工序的工艺参数 使用包含的层次关系。

\*/

//工艺术路线对象

@Transient

**private** CraRoute craRoute;

//工艺路线的工序列表

@Transient

**private** List<CraRouteSeq> seqLst = **new** ArrayList<CraRouteSeq>();

**CraRoute工艺路线对象：**

@Id

@Column(name="id")

@SequenceGenerator(name = "seq\_cra\_route", allocationSize = 1, initialValue = 1, sequenceName = "seq\_cra\_route")

@GeneratedValue(generator = "seq\_cra\_route", strategy = GenerationType.*SEQUENCE*)

**private** Integer id;

@Column(name="route\_code")

**private** String routeCode;

@Column(name="route\_name")

**private** String routeName;

@Column(name="create\_by")

**private** String createBy;

@Column(name="create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern="yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using=JsonDateSerializer.**class**)

**private** Date createDate;

**CraRouteSeq工序对象**

@Id

@Column(name="id")

@SequenceGenerator(name = "seq\_car\_route\_seq", allocationSize = 1, initialValue = 1, sequenceName = "seq\_car\_route\_seq")

@GeneratedValue(generator = "seq\_car\_route\_seq", strategy = GenerationType.*SEQUENCE*)

**private** Integer id;

@Column(name="route\_code")

**private** String routeCode;

@Column(name="seq\_code")

**private** String seqCode;

@Column(name="seq\_name")

**private** String seqName;

@Column(name = "sort")

**private** Integer sort;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date createDate;

@Column(name = "route\_seq\_code")

**private** String routeSeqCode;

//当前工序的工艺参数

@Transient

**private** List<CraSeqParam> cspLst = **new** ArrayList<CraSeqParam>();

**CraSeqParam一工序对应的生产参数，需收集**

@Id

@Column(name = "id")

@SequenceGenerator(name = "seq\_cra\_seq\_param", sequenceName = "seq\_cra\_seq\_param")

@GeneratedValue(generator = "seq\_cra\_seq\_param", strategy = GenerationType.*SEQUENCE*)

**private** Integer id;

@Column(name = "seq\_code")

**private** String seqCode;

@Column(name = "param\_code")

**private** String paramCode; //参数编码

**//参数部新需求说明。 系统添加目前想到的所有参数，即使没有value，把参数编码先传给机台。 设定、上、下限为空的参数，机台需要收集数据，回写数据时给出参考值，写入数据库**

**private** Double paramValue; //设定值 新增

**private** Double paramMinValue;

**private** Double paramMaxValue;

@Column(name = "uint")

**private** String uint;

@Column(name = "current\_seq\_ggxh")

**private** String currentSeqGgxh;

@Column(name = "craft\_seq\_code")

**private** String craftSeqCode;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

@Temporal(TemporalType.*TIMESTAMP*)

@DateTimeFormat(pattern = "yyyy-MM-dd HH:mm:ss")

@JsonSerialize(using = JsonDateSerializer.**class**)

**private** Date createDate;

//产品规格型号

@Column(name = "ggxh")

**private** String ggxh;

# 2机台上报生产异常

单个对象的json字符串。类结构：标出的字段必填。

**private** String code; **//异常编码 (后续异常编码做成参数，两边一致)**

**private** String macCode; **//机台编码**

**private** String meInfo; **//异常信息 200长度**

//private Date meTime;

**private** Long meTime; **//异常发生时间**

**private String seqCode; //工序编码**

**private** String axisName; **//出异常的轴名称**

**private** String exceptionParam; **//出现异常的参数编码（编码后续统一。） 有值时填**

**private** String exceptionValue; **//异常值 有值时填**

private String agentBy; **//操作人**

**private String courseCode; //当前工单**

**private String proCraftCode; // 产品工艺编码**

# 3机台打包上传数据---参数采集部分

**private** Integer machineId; //机台ID

**private** String axisName; //轴名称

//private String paramCode; //参数编码

//private String paramValue; //参数值

//private String isException; //是否异常（是或或）

//private Date cjTime; //采集时间点 Long

**private** Integer courseCode; //工单编号

**private** String seqCode; //工序编号

**private** String gzipParamInfoList; //压缩后的参数

# 4. 机台打包上传数据（有用的部分） 包括轴完成与参数回写

结构：

Obj

List< ParamVo >

//核心参数

//机台编码

**private** String macCode;

//轴名称

**private** String axisName;

//工序编码

**private** String seqCode;

//收线时间

//private Timestamp sxTime;

**private** Long sxTime;

//完成状态(正常结束；异常结束；)

**private** String status;

//放线轴rfid,多个以英文逗号隔开

**private** String rfids\_begin;

//private Timestamp actualBeginTime; //实际生产开始时间 从准备时间开始算起

//private Timestamp actualEndTime ; //实际生产结束时间 从落轴停机开始算起

**private** Long actualBeginTime;

**private** Long actualEndTime;

//实际物料配送到位时间 叉车配料或人手推。 这个字段以后需要叉车或手持机告诉你。现在可以随意给个值

//private Timestamp acutalDispatchTime;

**private** Long acutalDispatchTime;

//收线rfid. 收线只有一个轴

**private** String rfid\_end;

//本根线实际生产速度——平均

**private** BigDecimal product\_speed\_avg;

//参数回填部分

**private** List<ParamVo> lst = **new** ArrayList<ParamVo>();

ParamVo:

/\*\* 参数名称 \*/

**private** String param;

/\*\* 参数数值 \*/

//机台打包提交时，回填；设定值、最大、最小

**private** String value;

**private** String valueMax;

**private** String valueMin;

//参数状态（发送到机台时不填。 从机台回写时：要么没数据，要么发回的数据是必须要更新到数据库的；

//flag:更新; 不更新） 必须要更新参数的：flag=更新

**private** String flag;

# 5.机台向web发送呼叫叉车指令.及叉车回应

机台生产结束前10分钟，向web系统提出呼叫叉车的指令。格式如下

本格式涉及web端的监听机台指令p2p队列to杨超：(QUE\_WEB\_Command\_Receive\_p2p)，叉车端监听指令的主题订阅(to闻、刘) (QUE\_Topic\_Persist\_chaChe)二队列。

**private** Integer id; // 机台端不填 。 叉车app端撤销指令时要填。 Web向叉车发任务时要填

//保留字段，只： 叉车端(app)用于发消息取消任务。机台与web端不填

**private** String commType;

//工单号 机台、web要填

**private** String courseCode;

//仓库呼叫则不填。 机台端要填

**private** String fromMachineCode;

//仓库呼叫的填（闻） 要求，根据工单、机台、物料名称、规格型号分组合计出数量与单位: 只包含这6个字段，再加机台名称共7字段

//从仓库配料可能要配多种料到多个机台。所以用list

**private** List<PlaMachinePlanMater> materList;

//机台端不填 、web端填（翟），web端（闻）不填.

**private** String toMachineName;

//叉车应答 叉车的编号。 只叉车应答时填。其它时间不填

//叉车回应时，只填两个字段：id与chaCheCode

**private** String chaCheCode; //叉车的编码

其中：materList的结构: 机台端无此数据，直接为null即可

**private** Integer id;

@Column(name = "machine\_plan\_id")

**private** Integer machinePlanId;

@Column(name = "mater\_id")

**private** Integer materId;

@Column(name = "amount")

**private Double amount;**

@Column(name = "unit")

**private String unit;**

@Column(name = "ggxh")

**private String ggxh;**

@Column(name = "axis\_name")

**private** String axisName;

@Column(name = "machine\_id")

**private Integer machineId;**

@Column(name = "course\_code")

**private String courseCode;**

@Column(name = "plan\_send\_time")

**private** Timestamp planSendTime;

@Column(name = "create\_by")

**private** String createBy;

@Column(name = "create\_date")

**private** Date createDate;

@Column(name = "flag")

**private** String flag;

@Column(name = "send\_state")

**private** String sendState;

@Column(name = "mater\_name")

**private String materName;**

@Transient

**private String macName;**